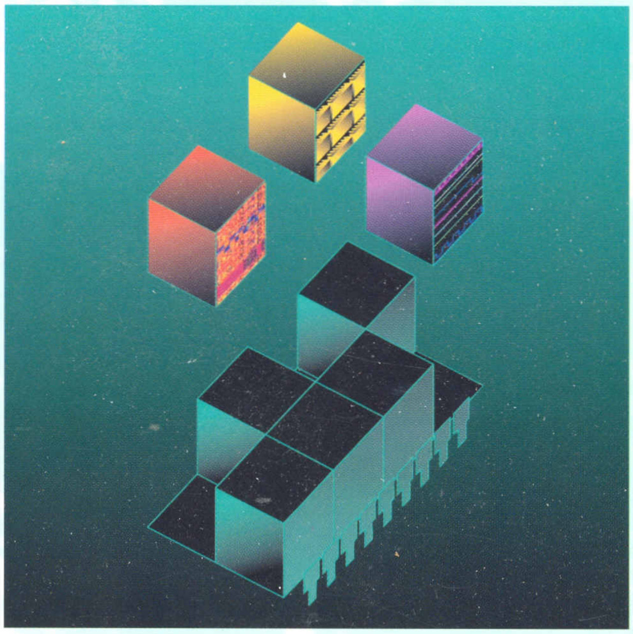


CONFIGURABLE LOGIC

DESIGN & APPLICATION BOOK

PLD • FPGA • GATE ARRAY

1994 • 1995

ATMEL

Atmel Programmable Logic Devices (PLDs)

PLDs	Pins	Vcc	Technology	tpd	Icc(mA)	Page
AT22V10-	24	5	UV CMOS	25,30	55	1-63
AT22V10-	24	5	UV CMOS	15,20	90	1-63
AT22V10B-	24	5	UV CMOS	7.5,10,12	140	1-77
AT22V10L-	24	5	UV CMOS	20,25,30	12	1-63
AT22LV10-	24	3	UV CMOS	20,25,30	35	1-87
AT22LV10L-	24	3	UV CMOS	25,30,35	4	1-87
ATF22V10B-	24	5	EE CMOS	7.5,10,15,25	120	1-35
ATF22V10BL-	24	5	EE CMOS	10,15	10	1-35
ATF22V10BQ-	24	5	EE CMOS	10	55	1-45
ATF22V10BQL-	24	5	EE CMOS	15,25	12	1-45
AT18V8Z-	20	5	UV CMOS	25,30,35	0.1	1-47
ATF16V8B-	20	5	EE CMOS	7.5,10,15,25	120	1-3
ATF16V8BL-	20	5	EE CMOS	10,15	10	1-3
ATF16V8BQ-	20	5	EE CMOS	10	55	1-17
ATF16V8BQL-	20	5	EE CMOS	15,25	12	1-17
ATF20V8B-	20	5	EE CMOS	7.5,10,15,25	120	1-19
ATF20V8BL-	20	5	EE CMOS	10,15	10	1-19
ATF20V8BQ-	20	5	EE CMOS	10	55	1-33
ATF20V8BQL-	20	5	EE CMOS	15,25	12	1-33
ATV750-	24	5	UV CMOS	20,25,30,35	120	1-99
ATV750L-	24	5	UV CMOS	25,30	12	1-99
ATV750B-	24	5	UV CMOS	7.5,10,15,25	180	1-115
ATV750BL-	24	5	UV CMOS	10,15,25	4	1-115
ATV750BQ-	24	5	UV CMOS	15,25	90	1-129
ATV750BQL-	24	5	UV CMOS	15,25	4	1-129
ATLV750B-	24	3	UV CMOS	10	45	1-131
ATLV750BL-	24	3	UV CMOS	15	2	1-131
ATV2500H-	40	5	UV CMOS	25,30,35	160	1-133
ATV2500L-	40	5	UV CMOS	30,35,40	5	1-133
ATV2500B-	40	5	UV CMOS	12,15	160	1-149
ATV2500BL-	40	5	UV CMOS	15,20	2	1-149
ATV2500BQ-	40	5	UV CMOS	15,25	90	1-163
ATV2500BQL-	40	5	UV CMOS	20,25	2	1-163
ATLV2500B-	40	3	UV CMOS	15	110	1-165
ATLV2500BL-	40	3	UV CMOS	20	1	1-165
ATV5000-	68	5	UV CMOS	25,30,35	350	1-167
ATV5000L-	68	5	UV CMOS	30,35	40	1-167
ATV5100-	68	5	UV CMOS	25,30,35	350	1-179
ATV5100L-	68	5	UV CMOS	30,35	40	1-179

Atmel Field Programmable Gate Arrays (FPGAs)

FPGAs	Usable Gates	Vcc	Registers	I/O	Fmax	Page
AT6002	2000	5	1024	96	250 MHz	2-3
AT6003	3000	5	1600	120	250 MHz	2-3
AT6005	5000	5	3136	108	250 MHz	2-3
AT6010	10000	5	6400	173	250 MHz	2-3

See inside back cover for an index to Atmel CMOS gate arrays.

Atmel CMOS Gate Arrays

Device Number	Raw Gates	Routable Gates	Max Pin Count	Max I/O Pins	Gate Speed	Page
ATL Array Organization						
ATL4	4,100	2,600	68	60	375 ps	4-3
ATL10	10,000	6,500	124	116	375 ps	4-3
ATL20	22,000	12,000	144	136	375 ps	4-3
ATL40	40,000	22,000	180	168	375 ps	4-3
ATL60	57,000	30,000	224	208	375 ps	4-3
ATL75	72,000	38,000	256	236	375 ps	4-3
ATL100	95,000	50,000	292	262	375 ps	4-3
ATL130	131,000	67,000	338	308	375 ps	4-3
ATL160	157,000	80,000	360	320	375 ps	4-3
ATL C Array Organization - Fine Pad Pitch						
ATL7C	7,000	4,000	100	92	375 ps	4-3
ATL10C	10,000	6,000	120	112	375 ps	4-3
ATL15C	15,000	8,000	144	136	375 ps	4-3
ATL20C	22,000	12,000	160	152	375 ps	4-3
ATL35C	35,000	18,000	208	192	375 ps	4-3
ATL55C	55,000	29,000	256	236	375 ps	4-3
ATL75C	75,000	39,000	304	280	375 ps	4-3
ATL80 Array Organization						
ATL80/15	17,000	10,200	100	92	256 ps	4-15
ATL80/25	26,000	15,600	120	112	256 ps	4-15
ATL80/40	39,000	23,400	144	136	256 ps	4-15
ATL80/50	50,000	30,000	160	152	256 ps	4-15
ATL80/75	75,000	45,000	184	176	256 ps	4-15
ATL80/95	94,000	60,000	208	192	256 ps	4-15
ATL80/150	150,000	75,000	256	236	256 ps	4-15
ATL80/220	220,000	110,000	304	280	256 ps	4-15
ATL80/280	280,000	140,000	340	310	256 ps	4-15
ATL80/350	350,000	175,000	380	350	256 ps	4-15
ATL80/450	450,000	225,000	424	384	256 ps	4-15
ATL80/600	600,000	300,000	480	440	256 ps	4-15
ATLV Array Organization						
ATLV2	2,000	1,400	44	36	1.3 ns	4-27
ATLV3	3,000	1,600	68	60	1.3 ns	4-27
ATLV5	5,000	2,800	84	76	1.3 ns	4-27
ATLV7	7,000	4,400	100	92	1.3 ns	4-27
ATLV10	10,000	6,600	120	112	1.3 ns	4-27
ATLV15	15,000	8,000	144	136	1.3 ns	4-27
ATLV20	22,000	12,000	160	152	1.3 ns	4-27
ATLV35	35,000	18,000	208	192	1.3 ns	4-27

Atmel Corporation

Configurable Logic Design and Application Book 1994 • 1995

PLD • FPGA • Gate Array

is the registered trademark of Atmel Corporation,
2125 O'Nel Drive, San Jose, CA 95131

Important Notice

Atmel guarantees that its circuits will be free from defects of material and workmanship under normal use and service, and that these circuits will perform to current specifications in accordance with, and subject to, the Company's standard warranty which is detailed in Atmel's Purchasing Order Acknowledgment.

Atmel reserves the right to change devices or specifications detailed in this data book at any time without notice, and assumes no responsibility for any errors within this document. Atmel does not make any commitment to update this information. Atmel assumes no responsibility for the use of any circuits described in this data book, nor does the Company assume responsibility for the functioning of undescribed features or parameters.

In the absence of a written agreement to the contrary, Atmel assumes no liability with respect to the use of semiconductor devices described in this data book for applications assistance, customers' product design or infringement of patents or copyrights of third parties.

Atmel's products are not authorized for use as critical components in life support devices or systems and the use as such implies that user bears all risk of such use.

If Atmel is an approved vendor on a Standard Military Drawing (SMD), the Atmel similar part number specification is compliant with the SMD.

Trademarks or registered trademarks used in this document may be the property of others.

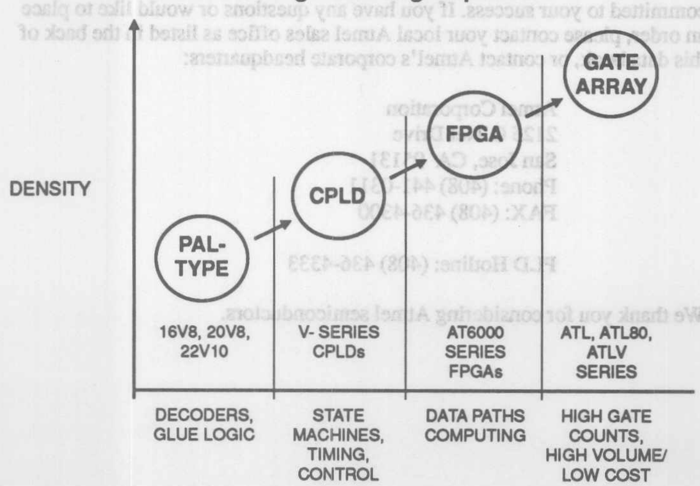
Printed on recycled paper.

Configurable Logic Overview

Thank you for your interest in Atmel's family of fast, flexible PLDs, FPGAs and Gate Arrays. Atmel offers the broadest range of configurable logic (programmable and custom ASIC, see figure) in the industry:

- Industry standard PAL- and GAL-type devices (16V8, 20V8, 22V10):
    - Low power
    - Very high speed
    - 3-volt and 5-volt operation
- High density CPLD devices (ATV750, ATV2500, ATV5000, ATV5100):
    - Up to 5,000 PLD gates using standard PLD design tools
    - Simple extensions of industry-standard 22V10 architecture
    - Wide gate fan-in capability
- FPGAs with *Cache Logic*™ capability, the ability to build adaptive, reusable hardware (AT6002, AT6003, AT6005, AT6010):
    - Up to 10,000 usable gates
    - Thousands of registers (ideal for pipelined, data path applications)
    - Flip-flop toggle rates up to 250 MHz, system speeds to 70 MHz
    - Very low power (500 µA standby)
    - Very low profile (PCMCIA) packages (1mm thick)
    - 100% factory-tested

Atmel's Configurable Logic Spectrum



- High density, high performance Gate Arrays:
    - Fast CMOS, 50 MHz and up
    - Low Voltage CMOS, 1.8-volt, 3.3-volt and 5-volt options
    - Gate Array/Standard Cell translation
    - Multiple FPGA/PLD conversions

If you are new to PLDs and FPGAs you'll appreciate the simple, easy-to-learn architecture and tools. If you are already using FPGAs you'll find the AT6000 FPGAs are especially suited for high speed, data path applications. The thousands of registers allow for designs to be extensively pipelined, enabling ultra-fast system throughput.

When your design becomes fixed and enters high volume production, the tools let you migrate easily to Atmel's masked Gate Arrays for even more cost effective production.

Atmel Corporation designs, manufactures, and markets high quality and high performance CMOS memory, logic and analog integrated circuits. Founded in 1984, the Company serves the manufacturers of computation, communications and instrumentation equipment in commercial, industrial and military environments.

Atmel's broad line of products provide customers with a variety of solutions to their memory and logic applications. Atmel offers high-density, high-speed memory and logic standard products as well as custom gate arrays.

Atmel guarantees quality and reliability by fabricating all products— no matter what their intended application— to meet or exceed the specifications of Military Standard 883.

Whether you are new to programmable logic or an experienced user, Atmel is committed to your success. If you have any questions or would like to place an order, please contact your local Atmel sales office as listed in the back of this data book, or contact Atmel's corporate headquarters:

Atmel Corporation
2125 O'Nel Drive
San Jose, CA 95131
Phone: (408) 441-0311
FAX: (408) 436-4300

PLD Hotline: (408) 436-4333

We thank you for considering Atmel semiconductors.

Table of Contents

Section 3 Programmable Logic Development Tools (continued)

Section 4 CMOS Gate Arrays

Section 5 PLD Application Notes

Section 9 Miscellaneous Information

Contents

1

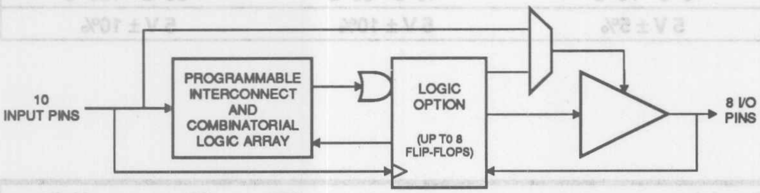
High Performance Flash PLD

Preliminary

Features

- **Industry Standard Architecture**
  - **Emulates Many 20-Pin PALs®**
  - **Low Cost Easy-to-Use Software Tools**
- **High Speed Electrically Erasable Programmable Logic Devices**
  - **7.5 ns Maximum Pin-to-Pin Delay**
- **Low Power ATF16V8BL - 10 mA Maximum Standby**
- **CMOS and TTL Compatible Inputs and Outputs**
  - **Input and I/O Pull-Up Resistors**
- **Advanced Flash Technology**
  - **Reprogrammable**
  - **100% Tested**
- **High Reliability CMOS Process**
  - **20 Year Data Retention**
  - **100 Erase/Write Cycles**
  - **2,000 V ESD Protection**
  - **200 mA Latchup Immunity**
- **Full Military, Commercial, and Industrial Temperature Ranges**
- **Dual-in-Line and Surface Mount Packages in Standard Pinouts**

Block Diagram



Description

The ATF16V8B is a high performance CMOS (Electrically Erasable) Programmable Logic Device (PLD) which utilizes Atmel's proven electrically erasable Flash memory technology. Speeds down to 7.5 ns and power dissipation as low as 10 mA are offered. All speed ranges are specified over the full 5 V ± 10% range for military and industrial temperature ranges, and 5 V ± 5% for commercial ranges.

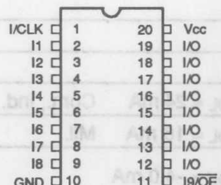
The ATF16V8BL provides the low power CMOS PLD solution, with low DC power (5.0 mA typical). The ATF16V8BL significantly reduces total system power and enhances system reliability

The ATF16V8Bs incorporate a superset of the generic architectures, which allows direct replacement of the 16R8 family and most 20-pin combinatorial PLDs. Eight outputs are each allocated eight product terms. Three different modes of operation, configured automatically with software, allow highly complex logic functions to be realized.

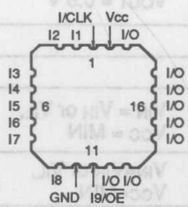
Pin Configurations

Pin Name	Function
CLK	Clock
I	Logic Inputs
I/O	Bidirectional Buffers
$\overline{OE}$	Output Enable
VCC	+5 V Supply

DIP



PLCC



Absolute Maximum Ratings*

Temperature Under Bias	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-2.0 V to +7.0 V(1)
Voltage on Input Pins with Respect to Ground During Programming	-2.0 V to +14.0 V(1)
Programming Voltage with Respect to Ground	-2.0 V to +14.0 V(1)

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Note:
1. Minimum voltage is -0.6 V dc, which may undershoot to -2.0 V for pulses of less than 20 ns. Maximum output pin voltage is Vcc + 0.75 V dc, which may overshoot to 7.0 V for pulses of less than 20 ns.

D.C. and A.C. Operating Conditions

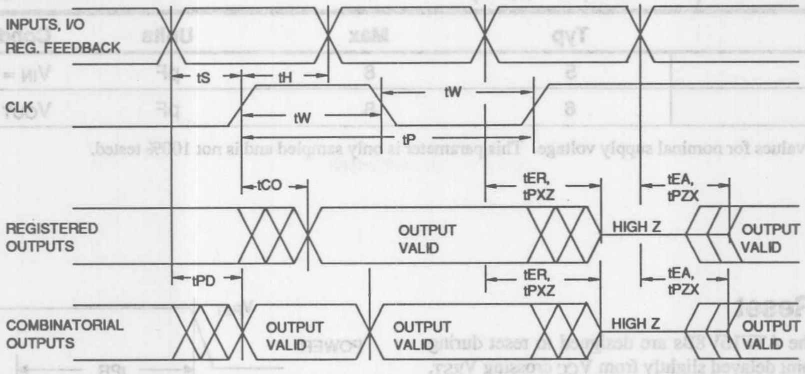
	Commercial	Industrial	Military
Operating Temperature (Case)	0°C - 70°C	-40°C - 85°C	-55°C - 125°C
V_{CC} Power Supply	5 V ± 5%	5 V ± 10%	5 V ± 10%

D.C. Characteristics

Symbol	Parameter	Condition			Min	Max	Units
I_{IL}	Input or I/O Low Leakage Current	$0 \leq V_{IN} \leq V_{IL}(MAX)$				-150	µA
I_{IH}	Input or I/O High Leakage Current	$3.5 \leq V_{IN} \leq V_{CC}$				10	µA
I_{CC}	Power Supply Current, Standby	V_{CC} = MAX, V_{IN} = MAX, Outputs Open	ATF16V8B	Com.		110	mA
				Ind., Mil.		120	mA
			ATF16V8BL	Com.		10	mA
				Ind., Mil.		15	mA
I_{CC2}	Clocked Power Supply Current	V_{CC} = MAX, Outputs Open	ATF16V8BL	Com.		15	mA/MHz(2)
				Ind., Mil.		20	mA/MHz(2)
I_{CC3}	Clocked Power Supply Current	V_{CC} = MAX, Outputs Open, f=25 MHz		Com.		115	mA
				Ind., Mil.		140	mA
I_{OS} (1)	Output Short Circuit Current	V_{OUT} = 0.5 V				-130	mA
V_{IL}	Input Low Voltage				-0.5	0.8	V
V_{IH}	Input High Voltage				2.0	V_{CC}+0.75	V
V_{OL}	Output Low Voltage	V_{IN} = V_{IH} or V_{IL}, V_{CC} = MIN	I_{OL} = 24 mA	Com., Ind.		0.5	V
			I_{OL} = 16 mA	Mil.		0.5	V
V_{OH}	Output High Voltage	V_{IN}=V_{IH} or V_{IL}, V_{CC}=MIN	I_{OH} = -4.0 mA		2.4		V

Notes: 1. Not more than one output at a time should be shorted. Duration of short circuit test should not exceed 30 sec.
2. Low frequency only, contact factory for I_{CC} versus frequency characterization curves.

A.C. Waveforms [(1)]



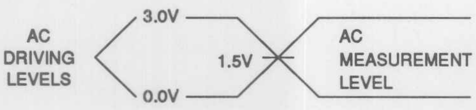
Note: 1. Timing measurement reference is 1.5 V. Input AC driving levels are 0.0 V and 3.0 V, unless otherwise specified.

A.C. Characteristics [(1)]

Symbol	Parameter	-7 Min	-7 Max	-10 Min	-10 Max	-15 Min	-15 Max	-25 Min	-25 Max	Units
t_{PD}	Input or Feedback to Non-Registered Output	3	7.5	3	10	3	15	3	25	ns
t_{CF}	Clock to Feedback		3		6		8		10	ns
t_{CO}	Clock to Output	2	5	2	7	2	10	2	12	ns
t_S	Input or Feedback Setup Time	5		7.5		12		15		ns
t_H	Hold Time	0		0		0		0		ns
t_P	Clock Period	8		12		16		24		ns
t_W	Clock Width	4		6		8		12		ns
F_{MAX}	External Feedback $1/(t_S+t_{CO})$		100		68		45		37	MHz
	Internal Feedback $1/(t_S+t_{CF})$		125		74		50		40	MHz
	No Feedback $1/(t_P)$		125		83		62		41	MHz
t_{EA}	Input to Output Enable — Product Term	3	9	3	10	3	15	3	25	ns
t_{ER}	Input to Output Disable — Product Term	2	9	2	10	2	15	2	25	ns
t_{PZX}	$\overline{OE}$ pin to Output Enable	2	6	2	10	2	15	2	20	ns
t_{PXZ}	$\overline{OE}$ pin to Output Disable	1.5	6	1.5	10	1.5	15	1.5	20	ns

Note: 1. See ordering information for valid part numbers and speed grades.

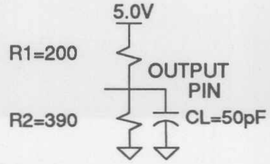
Input Test Waveforms and Measurement Levels:



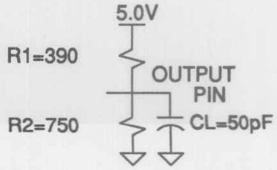
t_R, t_F < 5ns (10% to 90%)

Output Test Loads:

Commercial



Military



Pin Capacitance (f = 1 MHz, T = 25°C) [1]

	Typ	Max	Units	Conditions
C_{IN}	5	8	pF	V_{IN} = 0 V
C_{OUT}	6	8	pF	V_{OUT} = 0 V

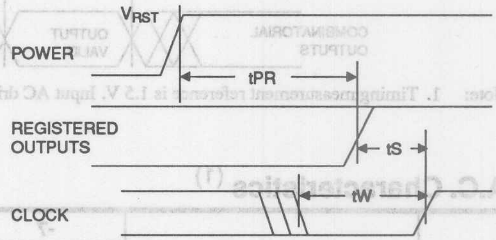
Note: 1. Typical values for nominal supply voltage. This parameter is only sampled and is not 100% tested.

Power Up Reset

The registers in the ATF16V8Bs are designed to reset during power up. At a point delayed slightly from V_{CC} crossing V_{RST}, all registers will be reset to the low state. As a result, the registered output state will always be high on power-up.

This feature is critical for state machine initialization. However, due to the asynchronous nature of reset and the uncertainty of how V_{CC} actually rises in the system, the following conditions are required:

1) The V_{CC} rise must be monotonic,
2) After reset occurs, all input and feedback setup times must be met before driving the clock term high, and
3) The signals from which the clock is derived must remain stable during t_{PR}.



Parameter	Description	Typ	Max	Units
t_{PR}	Power-Up Reset Time	600	1,000	ns
V_{RST}	Power-Up Reset Voltage	3.8	4.5	V

Preload of Registered Outputs

The ATF16V8B's registers are provided with circuitry to allow loading of each register with either a high or a low. This feature will simplify testing since any state can be forced into the registers to control test sequencing. A JEDEC file with preload is generated when a source file with vectors is compiled. Once downloaded, the JEDEC file preload sequence will be done automatically by most of the approved programmers after the programming.

Security Fuse Usage

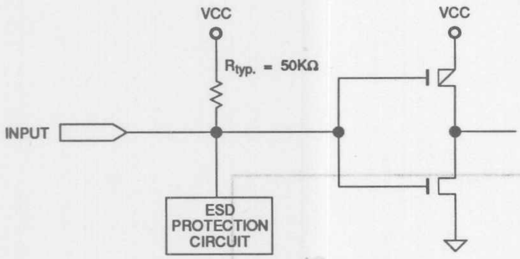
A single fuse is provided to prevent unauthorized copying of the ATF16V8B fuse patterns. Once programmed, fuse verify and preload are inhibited. However, the 64-bit User Signature remains accessible.

The security fuse should be programmed last, as its effect is immediate.

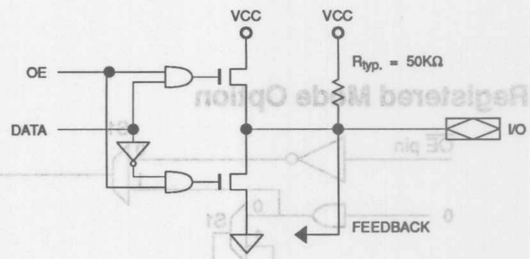
Input and I/O Pull-Ups

The ATF16V8B and ATF16V8BL have internal input and I/O active pull-up resistors. Therefore, whenever inputs or I/Os are not being driven externally, they will float to Vcc. This ensures that all logic array inputs are at known states. These are relatively weak active pull-ups that can easily be overdriven by TTL-compatible drivers (see input and I/O diagrams below).

Input Diagram



I/O Diagram



Functional Logic Diagram Description

The Logic Option and Functional Diagrams describe the ATF16V8B architecture. Eight configurable macrocells can be configured as a registered output, combinatorial I/O, combinatorial output, or dedicated input.

The ATF16V8B can be configured in one of three different modes. Each mode makes the ATF16V8B look like a different device. Most PLD compilers can choose the right mode automatically. The user can also force the selection by supplying the compiler with a mode selection. The determining factors would be the usage of register versus combinatorial outputs and dedicated outputs versus outputs with output enable control.

The ATF16V8B universal architecture can be programmed to emulate many 20-pin PAL devices. These architectural subsets can be found in each of the configuration modes described in the following pages. The user can download the listed subset device JEDEC programming file to the PLD programmer, and the ATF16V8B can be configured to act like the chosen device. Check with your programmer manufacturer for this capability.

Unused product terms are automatically disabled by the compiler to decrease power consumption. A Security Fuse, when programmed, protects the content of the ATF16V8B. Eight bytes (64 fuses) of User Signature are accessible to the user for purposes such as storing project name, part number, revision, or date. The User Signature is accessible regardless of the state of the Security Fuse.

Compiler Mode Selection

	Registered	Complex	Simple	Auto Select
ABEL, Atmel-ABEL	P16V8R	P16V8C	P16V8AS	P16V8
CUPL	G16V8MS	G16V8MA	G16V8AS	G16V8
LOG/IC	GAL16V8_R	GAL16V8_C7	GAL16V8_C8	GAL16V8
OrCAD-PLD	"Registered"	"Complex"	"Simple"	GAL16V8A
PLDesigner	P16V8R	P16V8C	P16V8C	P16V8A
Tango-PLD	G16V8R	G16V8C	G16V8AS	G16V8

ATF16V8B Registered Mode

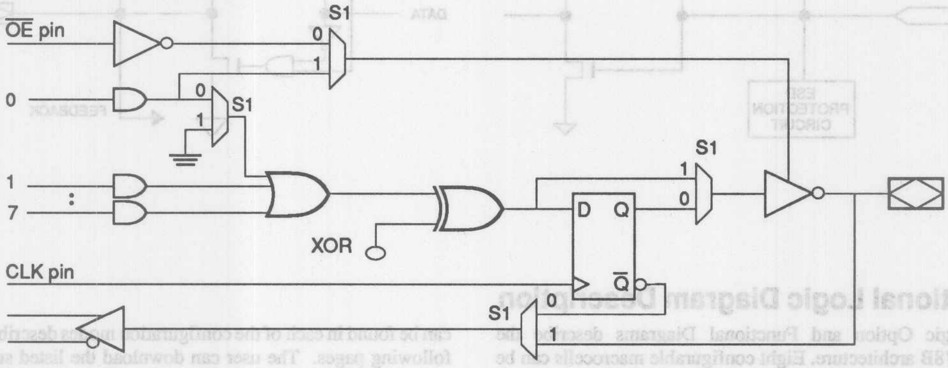
PAL Device Emulation / PAL Replacement

The registered mode is used if one or more registers are required. Each macrocell can be configured as either a registered or combinatorial output or I/O, or as an input. For a registered output or I/O, the output is enabled by the $\overline{OE}$ pin, and the register is clocked by the CLK pin. Eight product terms are allocated to the sum term. For a combinatorial output or I/O, the output enable is controlled by a product term, and seven product terms are allocated to the sum term. When the macrocell is configured as an input, the output enable is permanently disabled.

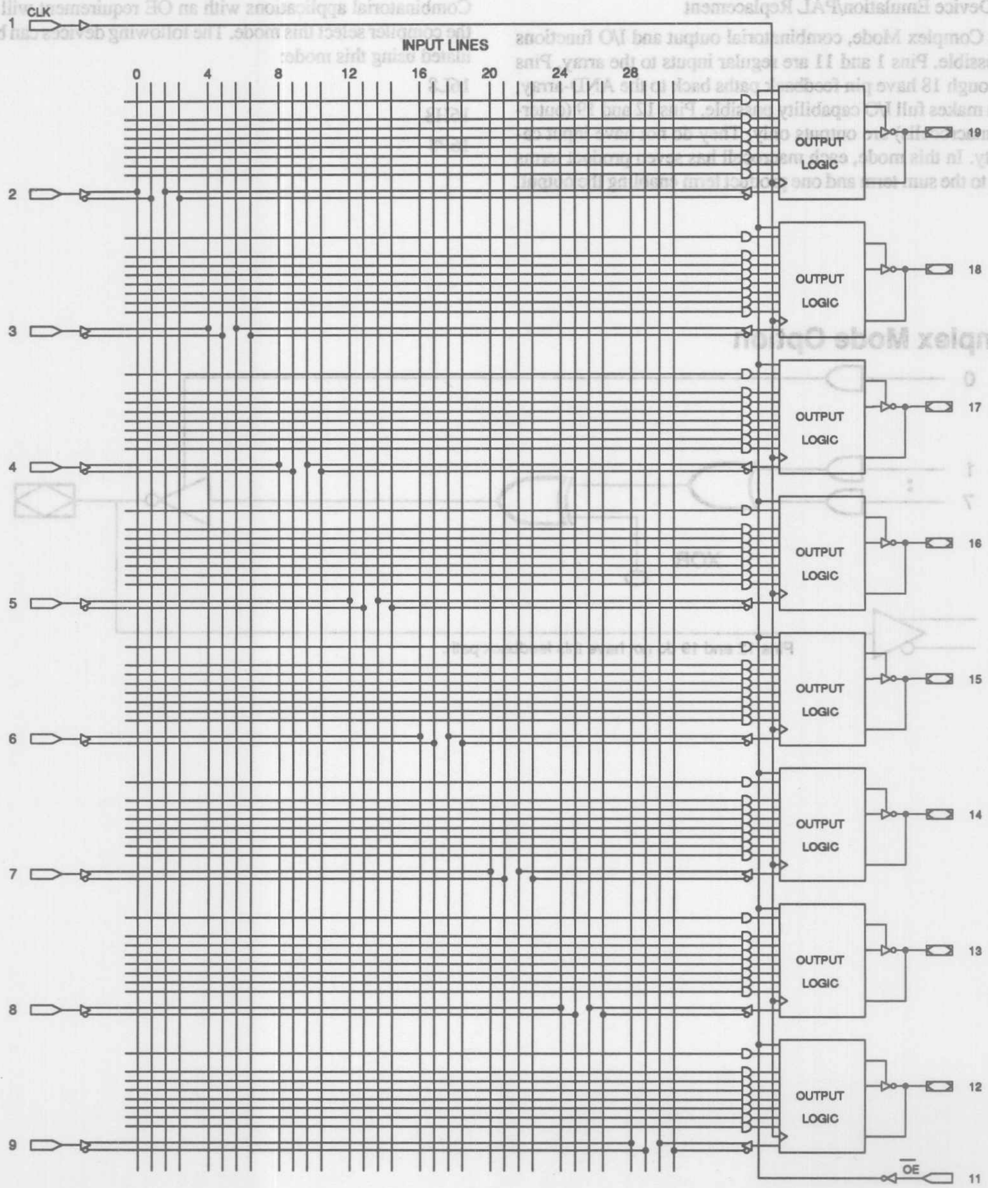
Any register usage will make the compiler select this mode. The following registered devices can be emulated using this mode:

16R8	16RP8
16R6	16RP6
16R4	16RP4

Registered Mode Option



Registered Mode Logic Diagram



ATF16V8B Complex Mode

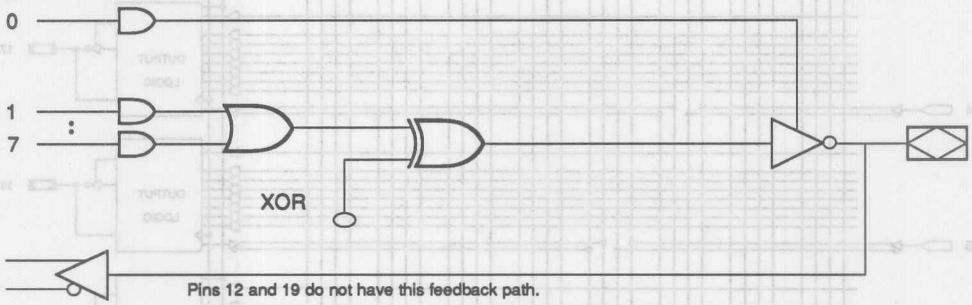
PAL Device Emulation/PAL Replacement

In the Complex Mode, combinatorial output and I/O functions are possible. Pins 1 and 11 are regular inputs to the array. Pins 13 through 18 have pin feedback paths back to the AND-array, which makes full I/O capability possible. Pins 12 and 19 (outermost macrocells) are outputs only. They do not have input capability. In this mode, each macrocell has seven product terms going to the sum term and one product term enabling the output.

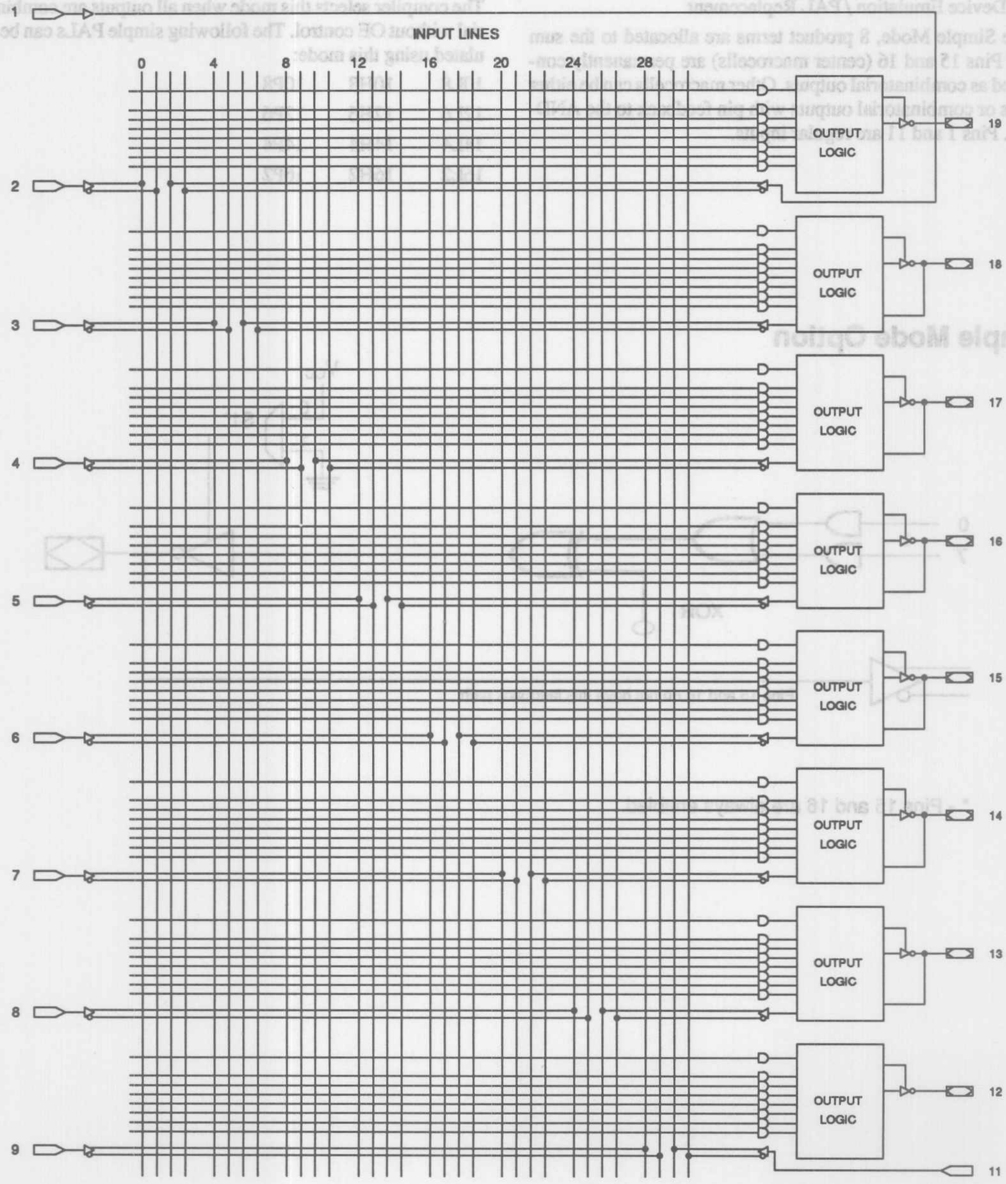
Combinatorial applications with an OE requirement will make the compiler select this mode. The following devices can be emulated using this mode:

16L8
16H8
16P8

Complex Mode Option



Complex Mode Logic Diagram



ATF16V8B Simple Mode

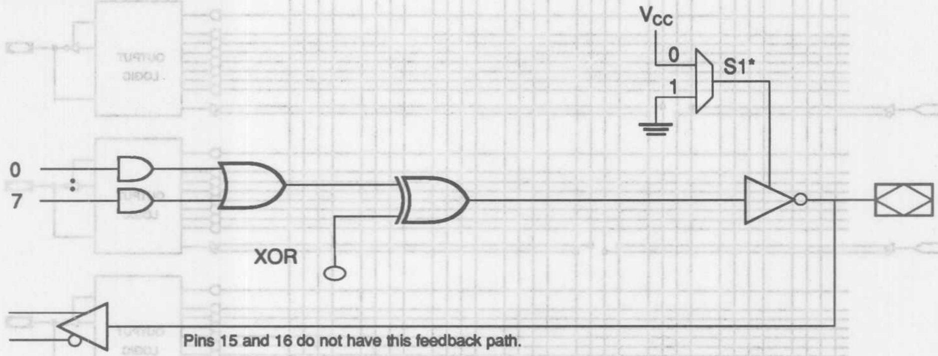
PAL Device Emulation / PAL Replacement

In the Simple Mode, 8 product terms are allocated to the sum term. Pins 15 and 16 (center macrocells) are permanently configured as combinatorial outputs. Other macrocells can be either inputs or combinatorial outputs with pin feedback to the AND-array. Pins 1 and 11 are regular inputs.

The compiler selects this mode when all outputs are combinatorial without OE control. The following simple PALs can be emulated using this mode:

10L8	10H8	10P8
12L6	12H6	12P6
14L4	14H4	14P4
16L2	16H2	16P2

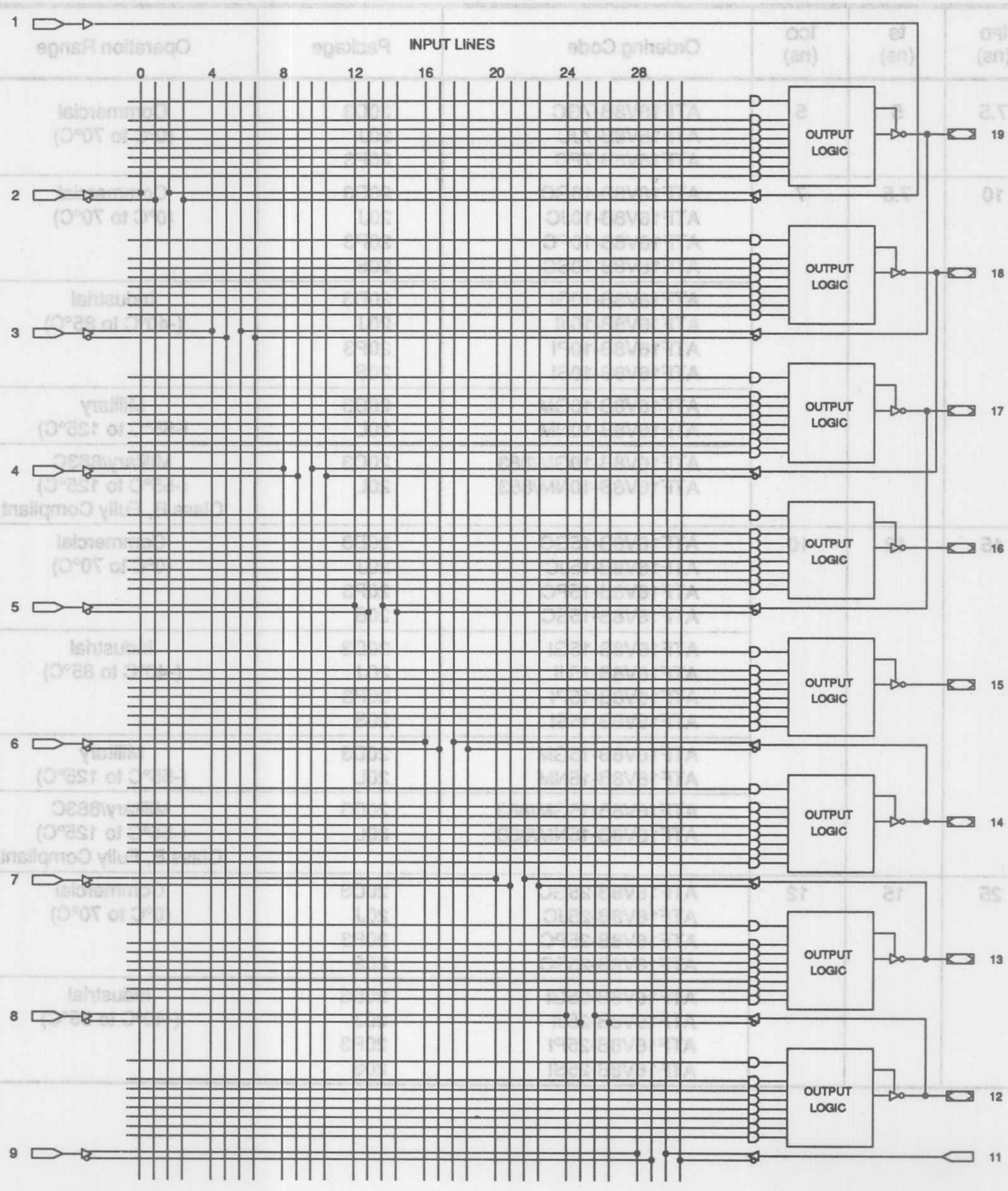
Simple Mode Option



* - Pins 15 and 16 are always enabled.

1

Simple Mode Logic Diagram



Ordering Information

t_{PD} (ns)	t_S (ns)	t_{CO} (ns)	Ordering Code	Package	Operation Range
7.5	5	5	ATF16V8B-7GC ATF16V8B-7JC ATF16V8B-7PC	20D3 20J 20P3	Commercial (0°C to 70°C)
10	7.5	7	ATF16V8B-10GC ATF16V8B-10JC ATF16V8B-10PC ATF16V8B-10SC	20D3 20J 20P3 20S	Commercial (0°C to 70°C)
			ATF16V8B-10GI ATF16V8B-10JI ATF16V8B-10PI ATF16V8B-10SI	20D3 20J 20P3 20S	Industrial (-40°C to 85°C)
			ATF16V8B-10GM ATF16V8B-10NM	20D3 20L	Military (-55°C to 125°C)
			ATF16V8B-10GM/883 ATF16V8B-10NM/883	20D3 20L	Military/883C (-55°C to 125°C) Class B, Fully Compliant
15	12	10	ATF16V8B-15GC ATF16V8B-15JC ATF16V8B-15PC ATF16V8B-15SC	20D3 20J 20P3 20S	Commercial (0°C to 70°C)
			ATF16V8B-15GI ATF16V8B-15JI ATF16V8B-15PI ATF16V8B-15SI	20D3 20J 20P3 20S	Industrial (-40°C to 85°C)
			ATF16V8B-15GM ATF16V8B-15NM	20D3 20L	Military (-55°C to 125°C)
			ATF16V8B-15GM/883 ATF16V8B-15NM/883	20D3 20L	Military/883C (-55°C to 125°C) Class B, Fully Compliant
25	15	12	ATF16V8B-25GC ATF16V8B-25JC ATF16V8B-25PC ATF16V8B-25SC	20D3 20J 20P3 20S	Commercial (0°C to 70°C)
			ATF16V8B-25GI ATF16V8B-25JI ATF16V8B-25PI ATF16V8B-25SI	20D3 20J 20P3 20S	Industrial (-40°C to 85°C)

Ordering Information

t_{PD} (ns)	t_S (ns)	t_{CO} (ns)	Ordering Code	Package	Operation Range
10	7.5	7	ATF16V8BL-10GC ATF16V8BL-10JC ATF16V8BL-10PC	20D3 20J 20P3	Commercial (0°C to 70°C)
			ATF16V8BL-10GI ATF16V8BL-10JI ATF16V8BL-10PI ATF16V8BL-10SI	20D3 20J 20P3 20S	Industrial (-40°C to 85°C)
15	12	10	ATF16V8BL-15GC ATF16V8BL-15JC ATF16V8BL-15PC ATF16V8BL-15SC	20D3 20J 20P3 20S	Commercial (0°C to 70°C)
			ATF16V8BL-15GI ATF16V8BL-15JI ATF16V8BL-15PI ATF16V8BL-15SI	20D3 20J 20P3 20S	Industrial (-40°C to 85°C)
			ATF16V8BL-15GM ATF16V8BL-15NM	20D3 20L	Military (-55°C to 125°C)
			ATF16V8BL-15GM/883 ATF16V8BL-15NM/883	20D3 20L	Military/883C (-55°C to 125°C) Class B, Fully Compliant

Package Type	
20D3	20 Lead, 0.300" Wide, Ceramic Dual Inline Package (Cerdip)
20J	20 Lead, Plastic J-Leaded Chip Carrier (PLCC)
20L	20 Pad, Ceramic Leadless Chip Carrier (LCC)
20P3	20 Lead, 0.300" Wide, Plastic Dual Inline Package (PDIP)
20S	20 Lead, 0.300" Wide, Plastic Gull Wing Small Outline (SOIC)

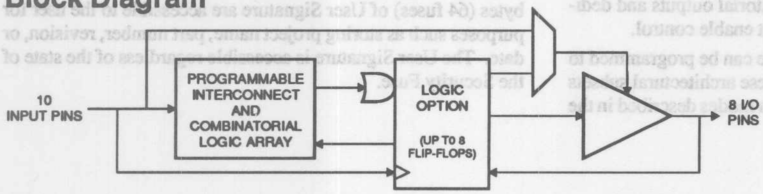
High Performance Flash PLD

Advanced Information

Features

- Quarter Power Equivalent of ATF16V8B - 55 mA Maximum
- Low Power ATF16V8BQL - 10 mA Maximum Standby
- Industry Standard Architecture
  - Emulates Many 20-Pin PALs®
  - Low Cost Easy-to-Use Software Tools
- High Speed Electrically Erasable Programmable Logic Devices
  - 10 ns Maximum Pin-to-Pin Delay
- CMOS and TTL Compatible Inputs and Outputs
  - Input and I/O Pull-Up Resistors
- Advanced Flash Technology
  - Reprogrammable
  - 100% Tested
- High Reliability CMOS Process
  - 20 Year Data Retention
  - 100 Erase/Write Cycles
  - 2,000 V ESD Protection
  - 200 mA Latchup Immunity
- Full Military, Commercial, and Industrial Temperature Ranges
- Dual-in-Line and Surface Mount Packages in Standard Pinouts

Block Diagram



Description

The ATF16V8BQs are high performance CMOS (Electrically Erasable) Programmable Logic Devices (PLDs) which utilize Atmel's proven electrically erasable Flash memory technology. Speeds down to 10 ns and power dissipation as low as 10 mA are offered. All speed ranges are specified over the full 5 V ± 10% range for military and industrial temperature ranges, and 5 V ± 5% for commercial ranges.

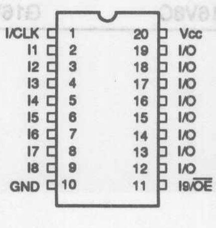
The ATF16V8BQL provides the low power CMOS PLD solution, with low DC power (5.0 mA typical). The ATF16V8BQL significantly reduces total system power and enhances system reliability

The ATF16V8BQs incorporate a superset of the generic architectures, which allows direct replacement of the 16R8 family and most 20-pin combinatorial PLDs. Eight outputs are each allocated eight product terms. Three different modes of operation, configured automatically with software, allow highly complex logic functions to be realized.

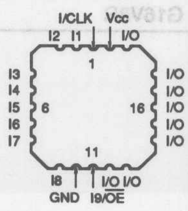
Pin Configurations

Pin Name	Function
CLK	Clock
I	Logic Inputs
I/O	Bidirectional Buffers
$\overline{OE}$	Output Enable
VCC	+5 V Supply

DIP



PLCC



D.C. and A.C. Operating Conditions

	Commercial	Industrial	Military
Operating Temperature (Case)	0^oC - 70^oC	-40^oC - 85^oC	-55^oC - 125^oC
V_{CC} Power Supply	5 V ± 5%	5 V ± 10%	5 V ± 10%

Functional Description

The ATF16V8BQ can be configured in one of three different modes. Each mode makes the ATF16V8BQ look like a different device. The ATF16V8BQ macrocells can be a registered output, combinatorial I/O, combinatorial output, or dedicated input. Most PLD compilers can choose the right mode automatically. The user can also force the selection by supplying the compiler with a mode selection. The determining factors would be the usage of register versus combinatorial outputs and dedicated outputs versus outputs with output enable control.

The ATF16V8BQ universal architecture can be programmed to emulate many 20-pin PAL devices. These architectural subsets can be found in each of the configuration modes described in the following pages. The user can download the listed subset device JEDEC programming file to the PLD programmer, and the ATF16V8BQ can be configured to act like the chosen device. Check with your programmer manufacturer for this capability.

Unused product terms are automatically disabled by the compiler to decrease power consumption. A Security Fuse, when programmed, protects the content of the ATF16V8BQ. Eight bytes (64 fuses) of User Signature are accessible to the user for purposes such as storing project name, part number, revision, or date. The User Signature is accessible regardless of the state of the Security Fuse.

Compiler Mode Selection

	Registered	Complex	Simple	Auto Select
ABEL, Atmel-ABEL	P16V8R	P16V8C	P16V8AS	P16V8
CUPL	G16V8MS	G16V8MA	G16V8AS	G16V8
LOG/iC	GAL16V8_R	GAL16V8_C7	GAL16V8_C8	GAL16V8
OrCAD-PLD	"Registered"	"Complex"	"Simple"	GAL16V8A
PLDesigner	P16V8R	P16V8C	P16V8C	P16V8A
Tango-PLD	G16V8R	G16V8C	G16V8AS	G16V8

1

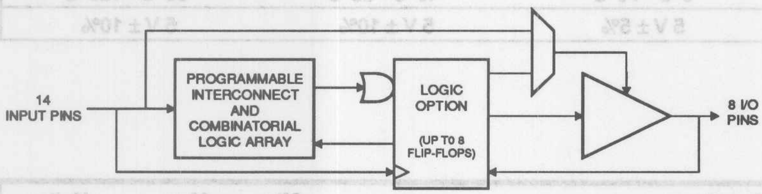
High Performance Flash PLD

Preliminary

Features

- **Industry Standard Architecture**
  - **Emulates Many 24-Pin PALs®**
  - **Low Cost Easy-to-Use Software Tools**
- **High Speed Electrically Erasable Programmable Logic Devices**
  - **7.5 ns Maximum Pin-to-Pin Delay**
- **Low Power ATF20V8BL - 10 mA Maximum Standby**
- **CMOS and TTL Compatible Inputs and Outputs**
  - **Input and I/O Pull-Up Resistors**
- **Advanced Flash Technology**
  - **Reprogrammable**
  - **100% Tested**
- **High Reliability CMOS Process**
  - **20 Year Data Retention**
  - **100 Erase/Write Cycles**
  - **2,000 V ESD Protection**
  - **200 mA Latchup Immunity**
- **Full Military, Commercial, and Industrial Temperature Ranges**
- **Dual-in-Line and Surface Mount Packages in Standard Pinouts**

Block Diagram



Description

The ATF20V8B is a high performance CMOS (Electrically Erasable) Programmable Logic Device (PLD) which utilizes Atmel's proven electrically erasable Flash memory technology. Speeds down to 7.5 ns and power dissipation as low as 10 mA are offered. All speed ranges are specified over the full 5 V ± 10% range for military and industrial temperature ranges, and 5 V ± 5% for commercial ranges.

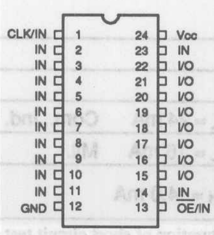
The ATF20V8BL provides the low power CMOS PLD solution, with low DC power (5.0 mA typical). The ATF20V8BL significantly reduces total system power and enhances system reliability

The ATF20V8Bs incorporate a superset of the generic architectures, which allows direct replacement of the 20R8 family and most 24-pin combinatorial PLDs. Eight outputs are each allocated eight product terms. Three different modes of operation, configured automatically with software, allow highly complex logic functions to be realized.

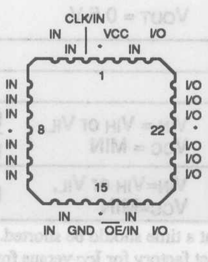
Pin Configurations

Pin Name	Function
CLK	Clock
IN	Logic Inputs
I/O	Bidirectional Buffers
$\overline{OE}$	Output Enable
*	No Internal Connection
VCC	+5 V Supply

DIP



PLCC



Absolute Maximum Ratings*

Temperature Under Bias	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-2.0 V to +7.0 V(1)
Voltage on Input Pins with Respect to Ground During Programming	-2.0 V to +14.0 V(1)
Programming Voltage with Respect to Ground	-2.0 V to +14.0 V(1)

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Note:

1. Minimum voltage is -0.6 V dc, which may undershoot to -2.0 V for pulses of less than 20 ns. Maximum output pin voltage is Vcc + 0.75 V dc, which may overshoot to 7.0 V for pulses of less than 20 ns.

D.C. and A.C. Operating Conditions

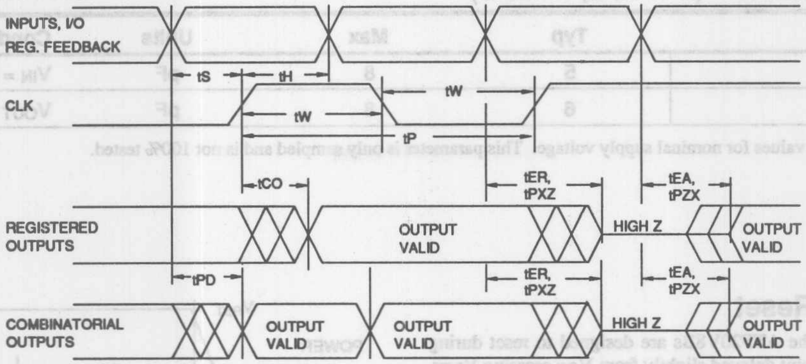
	Commercial	Industrial	Military
Operating Temperature (Case)	0°C - 70°C	-40°C - 85°C	-55°C - 125°C
V_{CC} Power Supply	5 V ± 5%	5 V ± 10%	5 V ± 10%

D.C. Characteristics

Symbol	Parameter	Condition			Min	Max	Units
I_{IL}	Input or I/O Low Leakage Current	$0 \le V_{IN} \le V_{IL}(MAX)$				-150	µA
I_{IH}	Input or I/O High Leakage Current	$3.5 \le V_{IN} \le V_{CC}$				10	µA
I_{CC}	Power Supply Current, Standby	V_{CC} = MAX, V_{IN} = MAX, Outputs Open	ATF20V8B	Com.		110	mA
				Ind., Mil.		120	mA
			ATF20V8BL	Com.		10	mA
				Ind., Mil.		15	mA
I_{CC2}	Clocked Power Supply Current	V_{CC} = MAX, Outputs Open	ATF20V8BL	Com.		15	mA/MHz(2)
				Ind., Mil.		20	mA/MHz(2)
I_{CC3}	Clocked Power Supply Current	V_{CC} = MAX, Outputs Open, f=25 MHz		Com.		115	mA
				Ind., Mil.		140	mA
I_{OS} (1)	Output Short Circuit Current	V_{OUT} = 0.5 V				-130	mA
V_{IL}	Input Low Voltage				-0.5	0.8	V
V_{IH}	Input High Voltage				2.0	V_{CC}+0.75	V
V_{OL}	Output Low Voltage	V_{IN} = V_{IH} or V_{IL}, V_{CC} = MIN	I_{OL} = 24 mA	Com., Ind.		0.5	V
			I_{OL} = 16 mA	Mil.		0.5	V
V_{OH}	Output High Voltage	V_{IN}=V_{IH} or V_{IL}, V_{CC}=MIN	I_{OH} = -4.0 mA		2.4		V

Notes:
1. Not more than one output at a time should be shorted. Duration of short circuit test should not exceed 30 sec.
2. Low frequency only, contact factory for I_{CC} versus frequency characterization curves.

A.C. Waveforms [1]



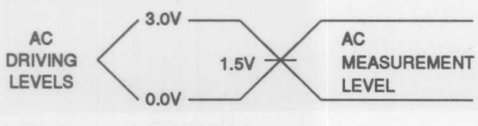
Note: 1. Timing measurement reference is 1.5 V. Input AC driving levels are 0.0 V and 3.0 V, unless otherwise specified.

A.C. Characteristics [1]

		-7		-10		-15		-25		
Symbol	**Parameter**	Min	Max	Min	Max	Min	Max	Min	Max	**Units**
t_{PD}	Input or Feedback to Non-Registered Output	3	7.5	3	10	3	15	3	25	ns
t_{CF}	Clock to Feedback		3		6		8		10	ns
t_{CO}	Clock to Output	2	5	2	7	2	10	2	12	ns
t_S	Input or Feedback Setup Time	5		7.5		12		15		ns
t_H	Hold Time	0		0		0		0		ns
t_P	Clock Period	8		12		16		24		ns
t_W	Clock Width	4		6		8		12		ns
F_{MAX}	External Feedback $1/(t_S+t_{CO})$		100		68		45		37	MHz
	Internal Feedback $1/(t_S + t_{CF})$		125		74		50		40	MHz
	No Feedback $1/(t_P)$		125		83		62		41	MHz
t_{EA}	Input to Output Enable — Product Term	3	9	3	10	3	15	3	25	ns
t_{ER}	Input to Output Disable — Product Term	2	9	2	10	2	15	2	25	ns
t_{PZX}	$\overline{OE}$ pin to Output Enable	2	6	2	10	2	15	2	20	ns
t_{PXZ}	$\overline{OE}$ pin to Output Disable	1.5	6	1.5	10	1.5	15	1.5	20	ns

Note: 1. See ordering information for valid part numbers and speed grades.

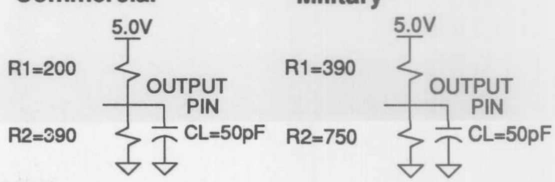
Input Test Waveforms and Measurement Levels:



t_R, t_F < 5ns (10% to 90%)

Output Test Loads:

Commercial **Military**

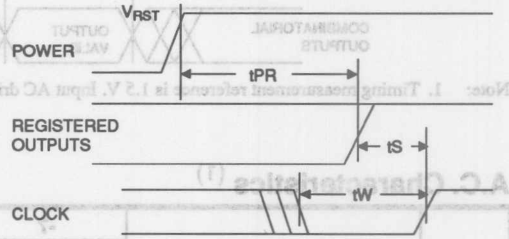


Pin Capacitance (f = 1 MHz, T = 25°C) [1]

	Typ	Max	Units	Conditions
C_{IN}	5	8	pF	V_{IN} = 0 V
C_{OUT}	6	8	pF	V_{OUT} = 0 V

Note: 1. Typical values for nominal supply voltage. This parameter is only sampled and is not 100% tested.

Power Up Reset



The registers in the ATF20V8Bs are designed to reset during power up. At a point delayed slightly from V_{CC} crossing V_{RST}, all registers will be reset to the low state. As a result, the registered output state will always be high on power-up.

This feature is critical for state machine initialization. However, due to the asynchronous nature of reset and the uncertainty of how V_{CC} actually rises in the system, the following conditions are required:

1) The V_{CC} rise must be monotonic,
2) After reset occurs, all input and feedback setup times must be met before driving the clock term high, and
3) The signals from which the clock is derived must remain stable during t_{PR}.

Parameter	Description	Typ	Max	Units
t_{PR}	Power-Up Reset Time	600	1,000	ns
V_{RST}	Power-Up Reset Voltage	3.8	4.5	V

Preload of Registered Outputs

The ATF20V8B's registers are provided with circuitry to allow loading of each register with either a high or a low. This feature will simplify testing since any state can be forced into the registers to control test sequencing. A JEDEC file with preload is generated when a source file with vectors is compiled. Once downloaded, the JEDEC file preload sequence will be done automatically by most of the approved programmers after the programming.

Security Fuse Usage

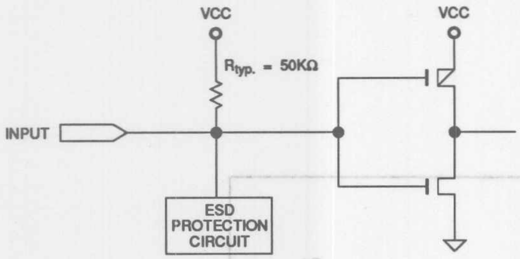
A single fuse is provided to prevent unauthorized copying of the ATF20V8B fuse patterns. Once programmed, fuse verify and preload are inhibited. However, the 64-bit User Signature remains accessible.

The security fuse should be programmed last, as its effect is immediate.

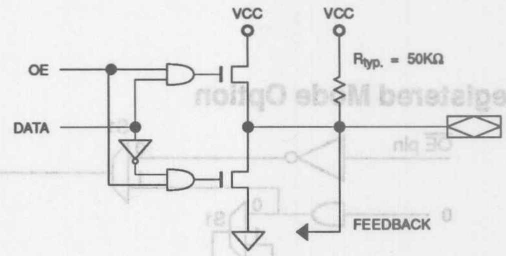
Input and I/O Pull-Ups

The ATF20V8B and ATF20V8BL have internal input and I/O active pull-up resistors. Therefore, whenever inputs or I/Os are not being driven externally, they will float to Vcc. This ensures that all logic array inputs are at known states. These are relatively weak active pull-ups that can easily be overdriven by TTL-compatible drivers (see input and I/O diagrams below).

Input Diagram



I/O Diagram



Functional Logic Diagram Description

The Logic Option and Functional Diagrams describe the ATF20V8B architecture. Eight configurable macrocells can be configured as a registered output, combinatorial I/O, combinatorial output, or dedicated input.

The ATF20V8B can be configured in one of three different modes. Each mode makes the ATF20V8B look like a different device. Most PLD compilers can choose the right mode automatically. The user can also force the selection by supplying the compiler with a mode selection. The determining factors would be the usage of register versus combinatorial outputs and dedicated outputs versus outputs with output enable control.

The ATF20V8B universal architecture can be programmed to emulate many 24-pin PAL devices. These architectural subsets can be found in each of the configuration modes described in the following pages. The user can download the listed subset device JEDEC programming file to the PLD programmer, and the ATF20V8B can be configured to act like the chosen device. Check with your programmer manufacturer for this capability.

Unused product terms are automatically disabled by the compiler to decrease power consumption. A Security Fuse, when programmed, protects the content of the ATF20V8B. Eight bytes (64 fuses) of User Signature are accessible to the user for purposes such as storing project name, part number, revision, or date. The User Signature is accessible regardless of the state of the Security Fuse.

Compiler Mode Selection

	Registered	Complex	Simple	Auto Select
ABEL, Atmel-ABEL	P20V8R	P20V8C	P20V8AS	P20V8
CUPL	G20V8MS	G20V8MA	G20V8AS	G20V8
LOG/IC	GAL20V8_R	GAL20V8_C7	GAL20V8_C8	GAL20V8
OrCAD-PLD	"Registered"	"Complex"	"Simple"	GAL20V8A
PLDesigner	P20V8R	P20V8C	P20V8C	P20V8A
Tango-PLD	G20V8R	G20V8C	G20V8AS	G20V8

ATF20V8B Registered Mode

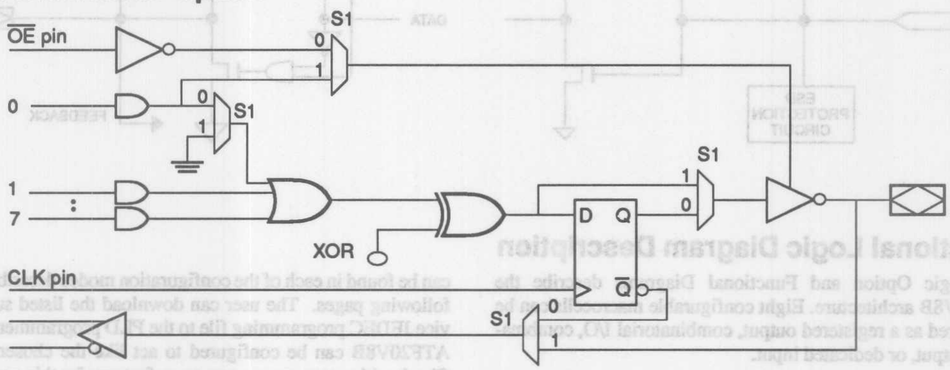
PAL Device Emulation / PAL Replacement

The registered mode is used if one or more registers are required. Each macrocell can be configured as either a registered or combinatorial output or I/O, or as an input. For a registered output or I/O, the output is enabled by the $\overline{OE}$ pin, and the register is clocked by the CLK pin. Eight product terms are allocated to the sum term. For a combinatorial output or I/O, the output enable is controlled by a product term, and seven product terms are allocated to the sum term. When the macrocell is configured as an input, the output enable is permanently disabled.

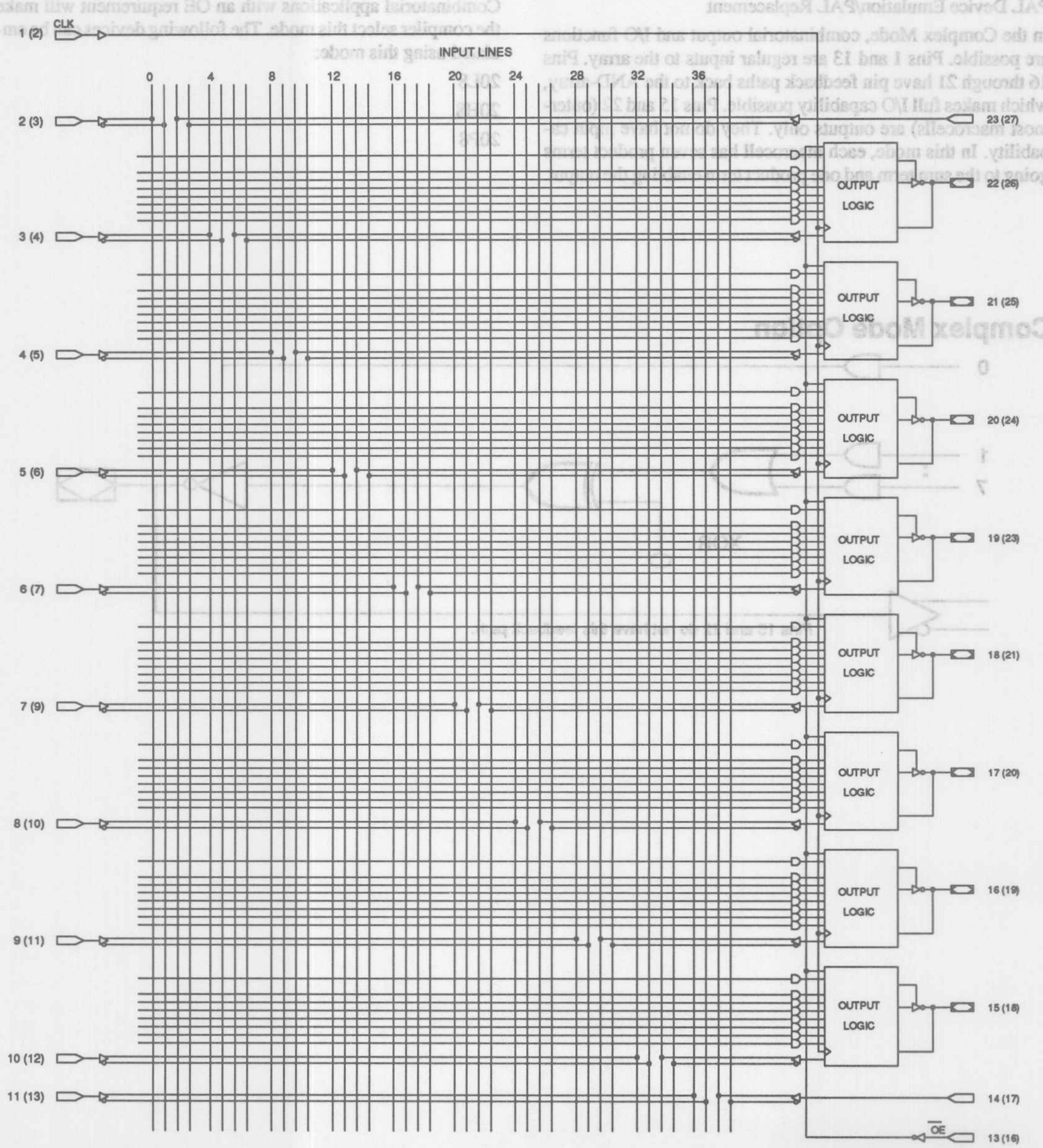
Any register usage will make the compiler select this mode. The following registered devices can be emulated using this mode:

20R8	20RP8
20R6	20RP6
20R4	20RP4

Registered Mode Option



Registered Mode Logic Diagram



ATF20V8B Complex Mode

PAL Device Emulation/PAL Replacement

In the Complex Mode, combinatorial output and I/O functions are possible. Pins 1 and 13 are regular inputs to the array. Pins 16 through 21 have pin feedback paths back to the AND-array, which makes full I/O capability possible. Pins 15 and 22 (outermost macrocells) are outputs only. They do not have input capability. In this mode, each macrocell has seven product terms going to the sum term and one product term enabling the output.

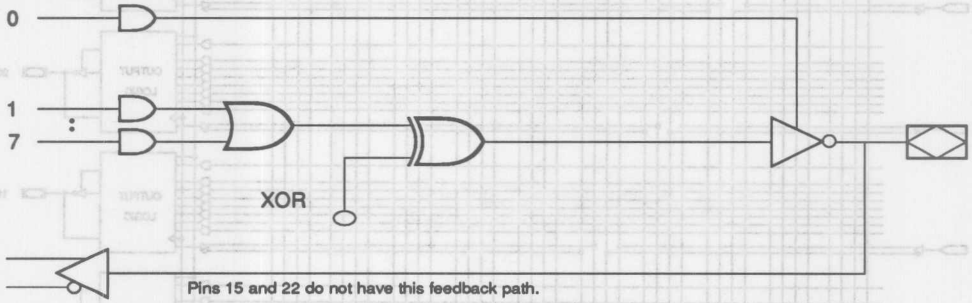
Combinatorial applications with an OE requirement will make the compiler select this mode. The following devices can be emulated using this mode:

20L8

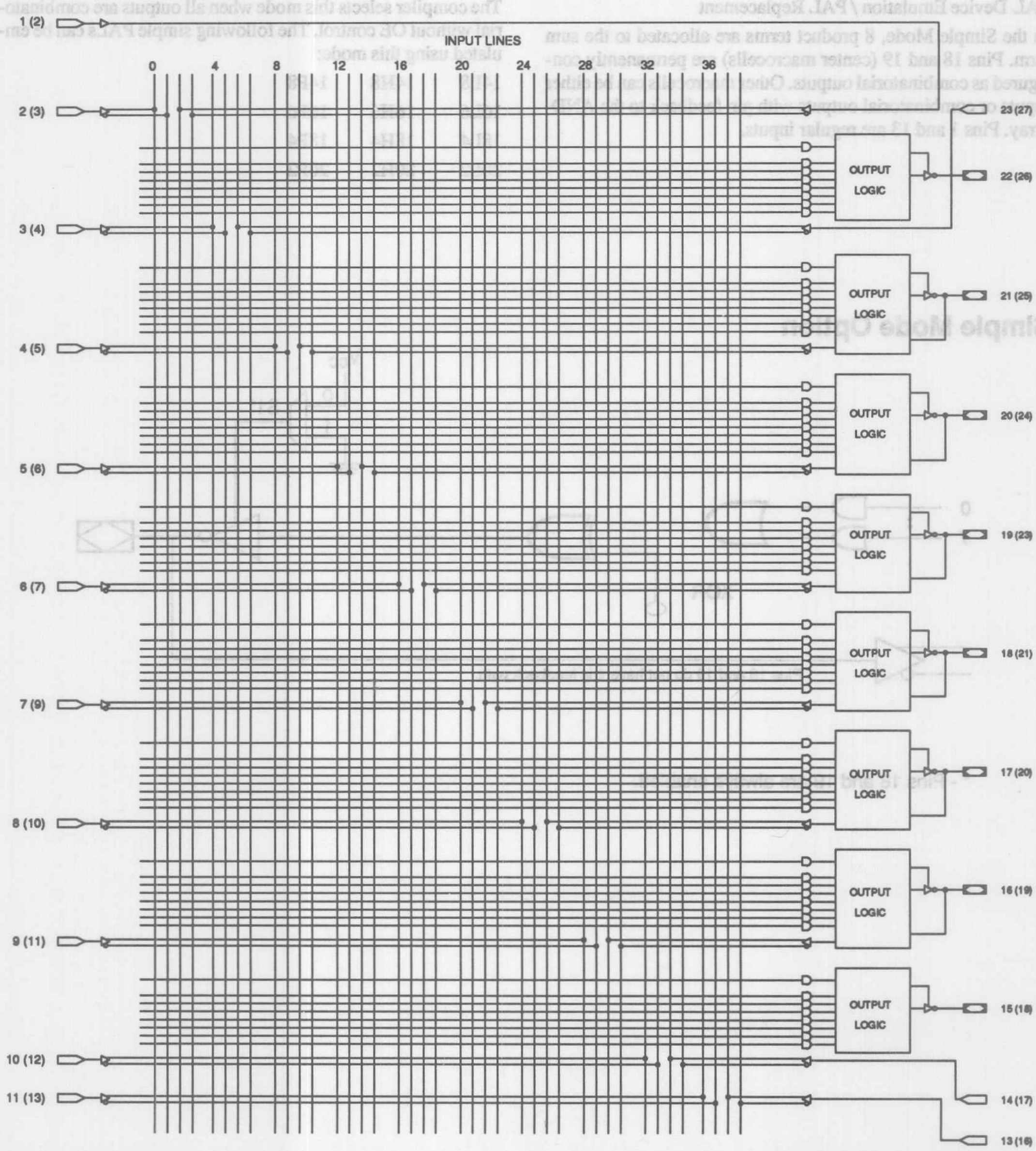
20H8

20P8

Complex Mode Option



Complex Mode Logic Diagram



ATF20V8B Simple Mode

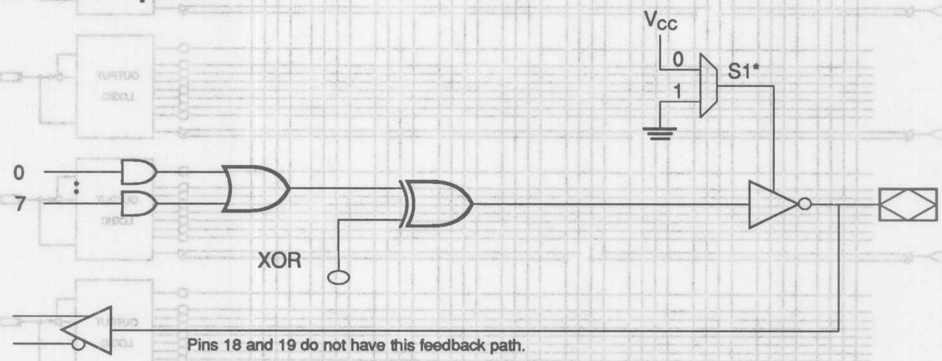
PAL Device Emulation / PAL Replacement

In the Simple Mode, 8 product terms are allocated to the sum term. Pins 18 and 19 (center macrocells) are permanently configured as combinatorial outputs. Other macrocells can be either inputs or combinatorial outputs with pin feedback to the AND-array. Pins 1 and 13 are regular inputs.

The compiler selects this mode when all outputs are combinatorial without OE control. The following simple PALs can be emulated using this mode:

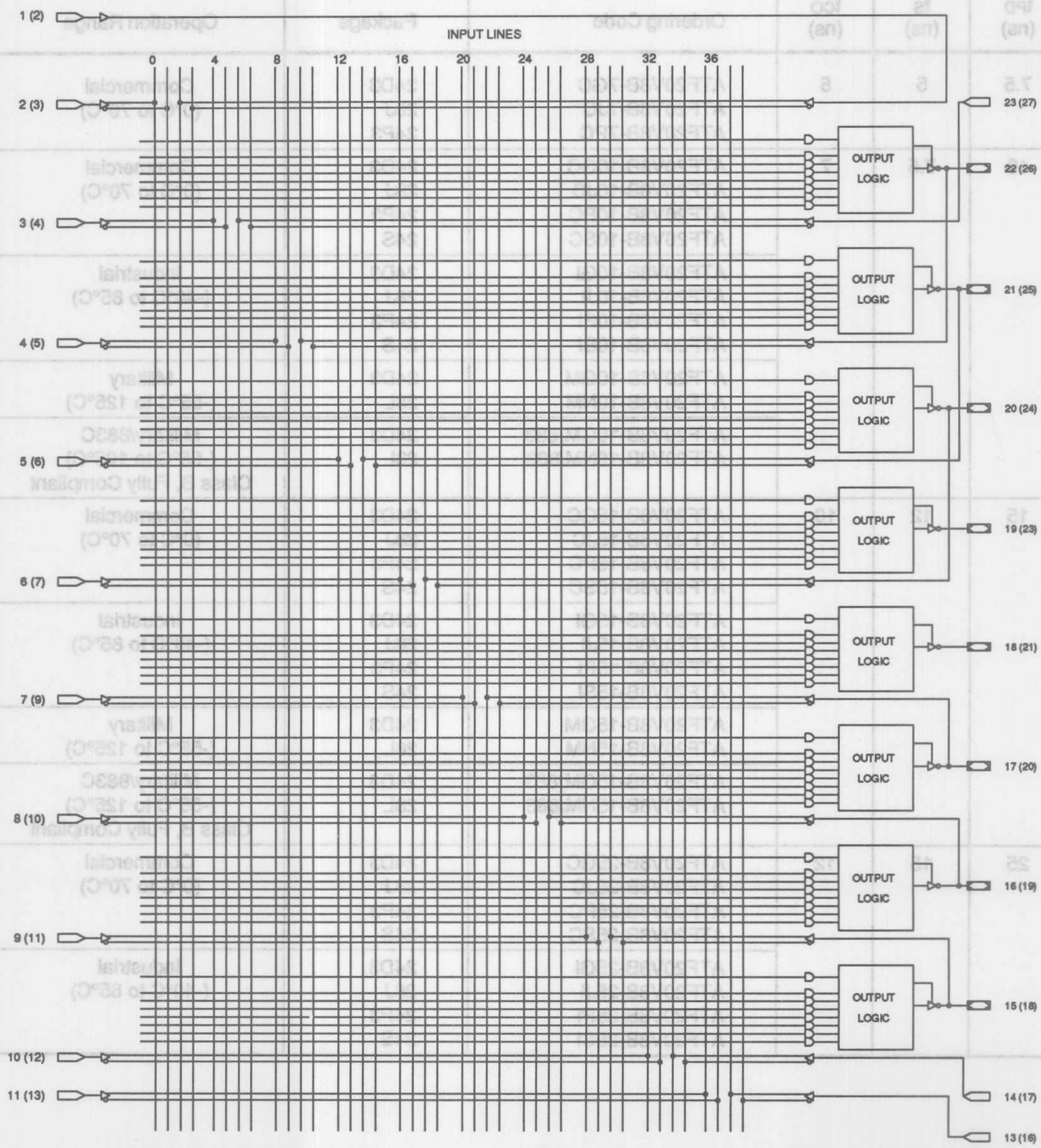
14L8	14H8	14P8
16L6	16H6	16P6
18L4	18H4	18P4
20L2	20H2	20P2

Simple Mode Option



* - Pins 18 and 19 are always enabled.

Simple Mode Logic Diagram



Ordering Information

t_{PD} (ns)	t_S (ns)	t_{CO} (ns)	Ordering Code	Package	Operation Range
7.5	5	5	ATF20V8B-7GC	24D3	Commercial
			ATF20V8B-7JC	28J	(0°C to 70°C)
			ATF20V8B-7PC	24P3	
10	7.5	7	ATF20V8B-10GC	24D3	Commercial
			ATF20V8B-10JC	28J	(0°C to 70°C)
			ATF20V8B-10PC	24P3	
			ATF20V8B-10SC	24S	
			ATF20V8B-10GI	24D3	Industrial
			ATF20V8B-10JI	28J	(-40°C to 85°C)
			ATF20V8B-10PI	24P3	
			ATF20V8B-10SI	24S	
			ATF20V8B-10GM	24D3	Military
			ATF20V8B-10NM	28L	(-55°C to 125°C)
			ATF20V8B-10GM/883	24D3	Military/883C
			ATF20V8B-10NM/883	28L	(-55°C to 125°C)
					Class B, Fully Compliant
15	12	10	ATF20V8B-15GC	24D3	Commercial
			ATF20V8B-15JC	28J	(0°C to 70°C)
			ATF20V8B-15PC	24P3	
			ATF20V8B-15SC	24S	
			ATF20V8B-15GI	24D3	Industrial
			ATF20V8B-15JI	28J	(-40°C to 85°C)
			ATF20V8B-15PI	24P3	
			ATF20V8B-15SI	24S	
			ATF20V8B-15GM	24D3	Military
			ATF20V8B-15NM	28L	(-55°C to 125°C)
			ATF20V8B-15GM/883	24D3	Military/883C
			ATF20V8B-15NM/883	28L	(-55°C to 125°C)
					Class B, Fully Compliant
25	15	12	ATF20V8B-25GC	24D3	Commercial
			ATF20V8B-25JC	28J	(0°C to 70°C)
			ATF20V8B-25PC	24P3	
			ATF20V8B-25SC	24S	
			ATF20V8B-25GI	24D3	Industrial
			ATF20V8B-25JI	28J	(-40°C to 85°C)
			ATF20V8B-25PI	24P3	
			ATF20V8B-25SI	24S	

1

Ordering Information

t_{PD} (ns)	t_S (ns)	t_{CO} (ns)	Ordering Code	Package	Operation Range
10	7.5	7	ATF20V8BL-10GC ATF20V8BL-10JC ATF20V8BL-10PC	24D3 28J 24P3	Commercial (0°C to 70°C)
			ATF20V8BL-10GI ATF20V8BL-10JI ATF20V8BL-10PI ATF20V8BL-10SI	24D3 28J 24P3 24S	Industrial (-40°C to 85°C)
15	12	10	ATF20V8BL-15GC ATF20V8BL-15JC ATF20V8BL-15PC ATF20V8BL-15SC	24D3 28J 24P3 24S	Commercial (0°C to 70°C)
			ATF20V8BL-15GI ATF20V8BL-15JI ATF20V8BL-15PI ATF20V8BL-15SI	24D3 28J 24P3 24S	Industrial (-40°C to 85°C)
			ATF20V8BL-15GM ATF20V8BL-15NM	24D3 28L	Military (-55°C to 125°C)
			ATF20V8BL-15GM/883 ATF20V8BL-15NM/883	24D3 28L	Military/883C (-55°C to 125°C) Class B, Fully Compliant

Package Type	
24D3	24 Lead, 0.300" Wide, Ceramic Dual Inline Package (Cerdip)
28J	28 Lead, Plastic J-Leaded Chip Carrier (PLCC)
28L	28 Pad, Ceramic Leadless Chip Carrier (LCC)
24P3	24 Lead, 0.300" Wide, Plastic Dual Inline Package (PDIP)
24S	24 Lead, 0.300" Wide, Plastic Gull Wing Small Outline (SOIC)

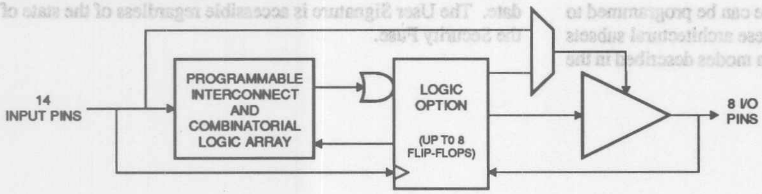
High Performance Flash PLD

Advanced Information

Features

- Quarter Power Equivalent of ATF20V8B - 55 mA Maximum
- Low Power ATF20V8BQL - 10 mA Maximum Standby
- Industry Standard Architecture
  - Emulates Many 24-Pin PALs®
  - Low Cost Easy-to-Use Software Tools
- High Speed Electrically Erasable Programmable Logic Devices
  - 10 ns Maximum Pin-to-Pin Delay
- CMOS and TTL Compatible Inputs and Outputs
  - Input and I/O Pull-Up Resistors
- Advanced Flash Technology
  - Reprogrammable
  - 100% Tested
- High Reliability CMOS Process
  - 20 Year Data Retention
  - 100 Erase/Write Cycles
  - 2,000 V ESD Protection
  - 200 mA Latchup Immunity
- Full Military, Commercial, and Industrial Temperature Ranges
- Dual-in-Line and Surface Mount Packages in Standard Pinouts

Block Diagram



Description

The ATF20V8BQs are high performance CMOS (Electrically Erasable) Programmable Logic Devices (PLDs) which utilize Atmel's proven electrically erasable Flash memory technology. Speeds down to 10 ns and power dissipation as low as 10 mA are offered. All speed ranges are specified over the full 5 V ± 10% range for military and industrial temperature ranges, and 5 V ± 5% for commercial ranges.

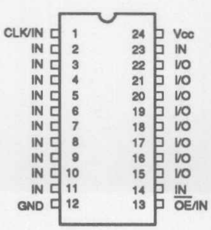
The ATF20V8BQL provides the low power CMOS PLD solution, with low DC power (5.0 mA typical). The ATF20V8BQL significantly reduces total system power and enhances system reliability

The ATF20V8BQs incorporate a superset of the generic architectures, which allows direct replacement of the 20R8 family and most 24-pin combinatorial PLDs. Eight outputs are each allocated eight product terms. Three different modes of operation, configured automatically with software, allow highly complex logic functions to be realized.

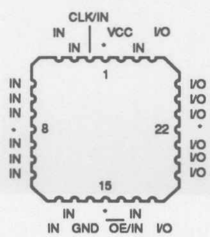
Pin Configurations

Pin Name	Function
CLK	Clock
IN	Logic Inputs
I/O	Bidirectional Buffers
$\overline{OE}$	Output Enable
*	No Internal Connection
VCC	+5 V Supply

DIP



PLCC



D.C. and A.C. Operating Conditions

	Commercial	Industrial	Military
Operating Temperature (Case)	0^oC - 70^oC	-40^oC - 85^oC	-55^oC - 125^oC
V_{CC} Power Supply	5 V ± 5%	5 V ± 10%	5 V ± 10%

Functional Description

The ATF20V8BQ can be configured in one of three different modes. Each mode makes the ATF20V8BQ look like a different device. The ATF20V8BQ macrocells can be a registered output, combinatorial I/O, combinatorial output, or dedicated input. Most PLD compilers can choose the right mode automatically. The user can also force the selection by supplying the compiler with a mode selection. The determining factors would be the usage of register versus combinatorial outputs and dedicated outputs versus outputs with output enable control.

The ATF20V8BQ universal architecture can be programmed to emulate many 24-pin PAL devices. These architectural subsets can be found in each of the configuration modes described in the following pages. The user can download the listed subset device JEDEC programming file to the PLD programmer, and the ATF20V8BQ can be configured to act like the chosen device. Check with your programmer manufacturer for this capability.

Unused product terms are automatically disabled by the compiler to decrease power consumption. A Security Fuse, when programmed, protects the content of the ATF20V8BQ. Eight bytes (64 fuses) of User Signature are accessible to the user for purposes such as storing project name, part number, revision, or date. The User Signature is accessible regardless of the state of the Security Fuse.

Compiler Mode Selection

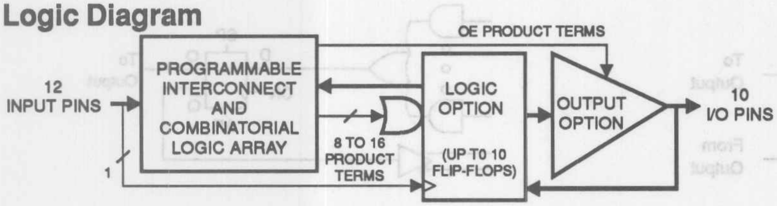
	Registered	Complex	Simple	Auto Select
ABEL, Atmel-ABEL	P20V8R	P20V8C	P20V8AS	P20V8
CUPL	G20V8MS	G20V8MA	G20V8AS	G20V8
LOG/IC	GAL20V8_R	GAL20V8_C7	GAL20V8_C8	GAL20V8
OrCAD-PLD	"Registered"	"Complex"	"Simple"	GAL20V8A
PLDesigner	P20V8R	P20V8C	P20V8C	P20V8A
Tango-PLD	G20V8R	G20V8C	G20V8AS	G20V8

High Performance Flash PLD

Features

- Industry Standard Architecture
  - Low-Cost, Easy-To-Use Software Tools
- High Speed Electrically Erasable Programmable Logic Device
  - 7.5 ns Max Propagation Delay
- Low Power ATF22V10BL - 10 mA Maximum Standby
- CMOS and TTL Compatible Inputs and Outputs
  - Input and I/O Pull-Up Resistors
- Advanced Flash Technology
  - Reprogrammable
  - 100% Tested
- High Reliability CMOS Technology
  - 20 Year Data Retention
  - 100 Erase/Write Cycles
  - 2,000 V ESD Protection
  - 200 mA Latchup Immunity
- Full Military, Commercial and Industrial Temperature Ranges
- Dual-In-Line and Surface Mount Packages in Standard Pinouts

Logic Diagram



Description

The ATF22V10B and ATF22V10BL are high performance CMOS (Electrically Erasable) Programmable Logic Devices (PLDs) which utilize Atmel's proven electrically erasable Flash memory technology. Speeds down to 7.5 ns and power dissipation as low as 10 mA are offered. All speed ranges are specified over the full 5 V ± 10% range for military and industrial temperature ranges, and 5V ± 5% for commercial ranges.

The ATF22V10BL provides the fastest low power CMOS PLD solution, with low DC power (5.0 mA typical). The ATF22V10BL significantly reduces total system power and enhances system reliability.

The ATF22V10B and ATF22V10BL incorporate a variable product term architecture. Each output is allocated from eight to 16 product terms, which allows highly complex logic functions to be realized.

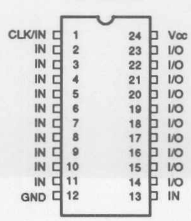
Two additional product terms are included to provide synchronous preset and asynchronous reset. These terms are common to all 10 registers. All registers are automatically cleared upon power up.

Register Preload simplifies testing. A Security Fuse prevents unauthorized copying of programmed fuse patterns.

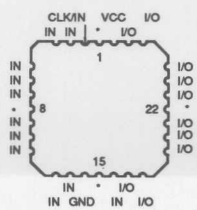
Pin Configurations

Pin Name	Function
CLK	Clock
IN	Logic Inputs
I/O	Bidirectional Buffers
*	No Internal Connection
VCC	+5 V Supply

DIP/SOIC



LCC/PLCC



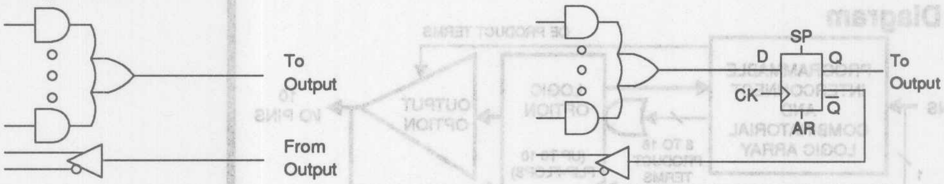
Absolute Maximum Ratings*

Temperature Under Bias	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-2.0 V to +7.0 V(1)
Voltage on Input Pins with Respect to Ground During Programming	-2.0 V to +14.0 V(1)
Programming Voltage with Respect to Ground	-2.0 V to +14.0 V(1)

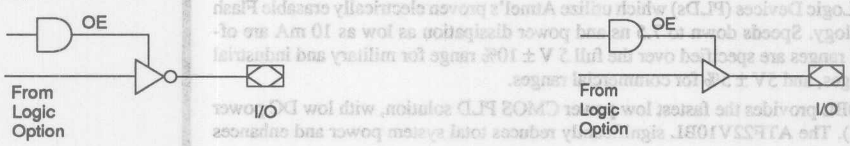
*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Note:
1. Minimum voltage is -0.6 V dc which may undershoot to -2.0 V for pulses of less than 20 ns. Maximum output pin voltage is V_{CC}+0.75 V dc which may overshoot to +7.0 V for pulses of less than 20 ns.

Logic Options



Output Options



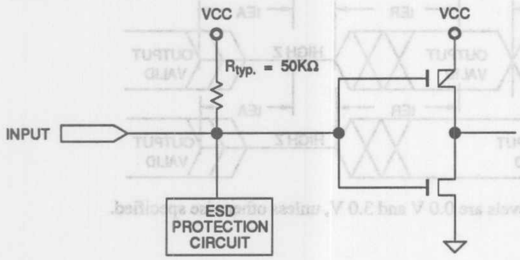
D.C. and A.C. Operating Conditions

	Commercial	Industrial	Military
Operating Temperature (Case)	0°C - 70°C	-40°C - 85°C	-55°C - 125°C
V_{CC} Power Supply	5 V ± 5%	5 V ± 10%	5 V ± 10%

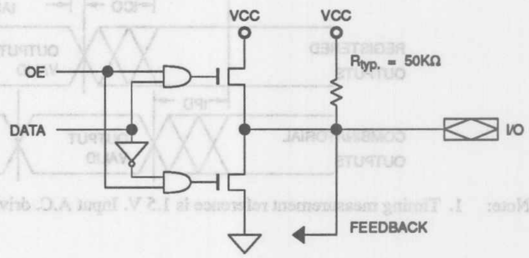
Input and I/O Pull-Ups

The ATF22V10B and ATF22V10BL have internal input and I/O active pull-up resistors. Therefore, whenever inputs or I/Os are not being driven externally, they will float to Vcc. This ensures that all logic array inputs are at known states. These are relatively weak active pull-ups that can easily be overdriven by TTL compatible drivers (see input and I/O diagrams below).

Input Diagram



I/O Diagram

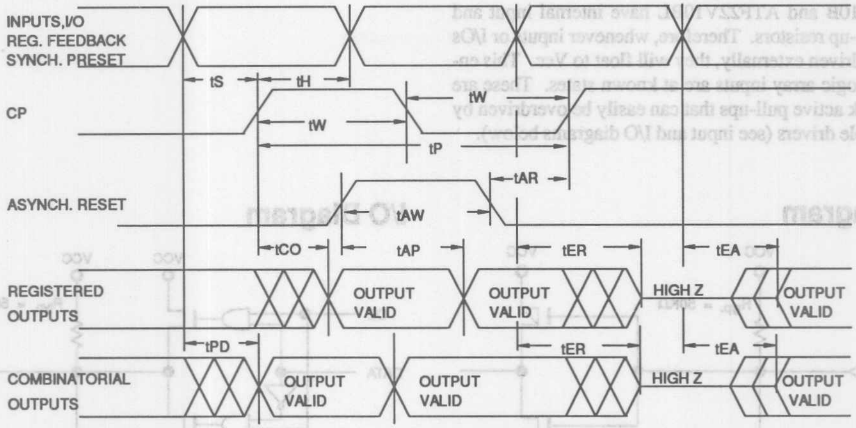


D.C. Characteristics

Symbol	Parameter	Condition			Min	Typ	Max	Units
I_{LI}	Input or I/O Low Leakage Current	$0 \le V_{IN} \le V_{IL}(MAX)$					150	µA
I_{LO}	Input or I/O High Leakage Current	$3.5 \le V_{IN} \le V_{CC}$					10	µA
I_{CC}	Power Supply Current, Standby	V_{CC} = MAX, V_{IN} = MAX, Outputs Open	ATF22V10B	Com.		90	120	mA
				Ind., Mil.		100	130	mA
			ATF22V10BL	Com.		5	10	mA
				Ind., Mil.		10	15	mA
I_{CC2}	Clocked Power Supply Current	V_{CC} = MAX, Outputs Open	ATF22V10BL	Com.			15	mA/MHz[2]
				Ind., Mil.			20	mA/MHz[2]
I_{CC3}	Clocked Power Supply Current	VCC = MAX, Outputs Open, f=25 MHz		Com.			130	mA
				Ind., Mil.			160	mA
I_{OS} [1]	Output Short Circuit Current	V_{OUT} = 0.5 V					-130	mA
V_{IL}	Input Low Voltage				-0.5		0.8	V
V_{IH}	Input High Voltage				2.0		V_{CC}+0.75	V
V_{OL}	Output Low Voltage	$V_{IN} = V_{IH}$ or V_{IL}, V_{CC} = MIN	I_{OL} = 16 mA	Com.,Ind.			0.5	V
			I_{OL} = 12 mA	Mil.			0.5	V
			I_{OL} = 24 mA	Com.			0.8	V
V_{OH}	Output High Voltage	$V_{IN}=V_{IH}$ or V_{IL}, V_{CC}=MIN	I_{OH} = -4.0 mA		2.4			V

Notes: 1. Not more than one output at a time should be shorted. Duration of short circuit test should not exceed 30 sec.
2. See I_{CC} versus frequency characterization curves.

A.C. Waveforms(1)



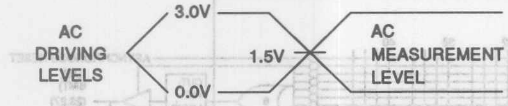
Note: 1. Timing measurement reference is 1.5 V. Input A.C. driving levels are 0.0 V and 3.0 V, unless otherwise specified.

A.C. Characteristics(1)

Symbol	Parameter	-7		B/BL-10		-15		-25		Units
		Min	Max	Min	Max	Min	Max	Min	Max	
t_{PD}	Input or Feedback to Combinatorial Output	3	7.5	3	10	3	15	3	25	ns
t_{CO}	Clock to Output	2	5	2	7	2	8	2	15	ns
t_{CF}	Clock to Feedback		3.5		4		4.5		13	ns
t_S	Input or Feedback Setup Time	3		4/7		10		15		ns
t_H	Hold Time	0		0		0		0		ns
F_{MAX}	External Feedback $1/(t_S + t_{CO})$	125		90/71.4		55.5		33.3		MHz
	Internal Feedback $1/(t_S + t_{CF})$	153		125/90		69		35.7		MHz
	No Feedback	166		125		83.3		38.5		MHz
t_P	Clock Period	6		8		12		26		ns
t_W	Clock Width	3		4		6		13		ns
t_{EA}	Input or I/O to Output Enable	3	7.5	3	10	3	15	3	25	ns
t_{ER}	Input or I/O to Output Disable	3	7.5	3	10	3	15	3	25	ns
t_{AP}	Input or I/O to Asynchronous Reset of Register	3	10	3	13	3	20	3	25	ns
t_{AW}	Asynchronous Reset Width	7		10		15		25		ns
t_{AR}	Asynchronous Reset Recovery Time	5		8		10		25		ns
t_{SP}	Setup Time, Synchronous Preset	4.5		6/10		10		15		ns
t_{SPR}	Synchronous Preset to Clock Recovery Time	5		10		10		15		ns

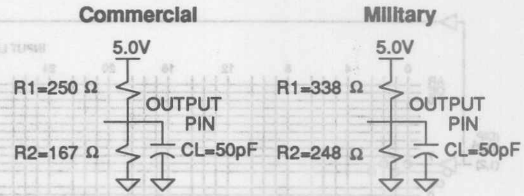
Note: 1. See ordering information for valid part numbers.

Input Test Waveforms and Measurement Levels



tR, tF < 3 ns (10% to 90%)

Output Test Loads:



Pin Capacitance (f = 1 MHz, T = 25°C) [1]

	Typ	Max	Units	Conditions
C_{IN}	5	8	pF	V_{IN} = 0 V
C_{OUT}	6	8	pF	V_{OUT} = 0 V

Note: 1. Typical values for nominal supply voltage. This parameter is only sampled and is not 100% tested.

Power Up Reset

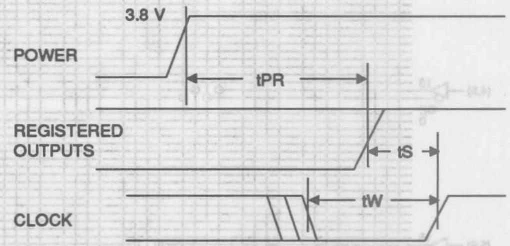
The registers in the ATF22V10B and ATF22V10BL are designed to reset during power up. At a point delayed slightly from V_{CC} crossing 3.8 V, all registers will be reset to the low state. The output state will depend on the polarity of the output buffer.

This feature is critical for state machine initialization. However, due to the asynchronous nature of reset and the uncertainty of how V_{CC} actually rises in the system, the following conditions are required:

1) The V_{CC} rise must be monotonic,

2) After reset occurs, all input and feedback setup times must be met before driving the clock pin high, and

3) The clock must remain stable during t_{PR}.



Parameter	Description	Min	Typ	Max	Units
t_{PR}	Power-Up Reset Time		600	1000	ns

Preload of Registered Outputs

When testing state machine designs, all possible states and state transitions must be verified in the design, not just those required in the normal machine operations. This is because certain events may occur during system operation that throw the logic into an illegal state (power-up, line voltage glitches, brown-outs, etc.). To test a design for proper treatment of these conditions, a way must be provided to break the feedback paths, and force any desired (i.e., illegal) state into the registers. Then the machine can be sequenced and the outputs tested for correct next state conditions.

The ATF22V10B/BL device includes circuitry that allows each registered output to be synchronously set either high or low. Thus, any present state condition can be forced for test sequencing. If necessary, approved programmers capable of executing test vectors perform output register preload automatically.

Device Programming

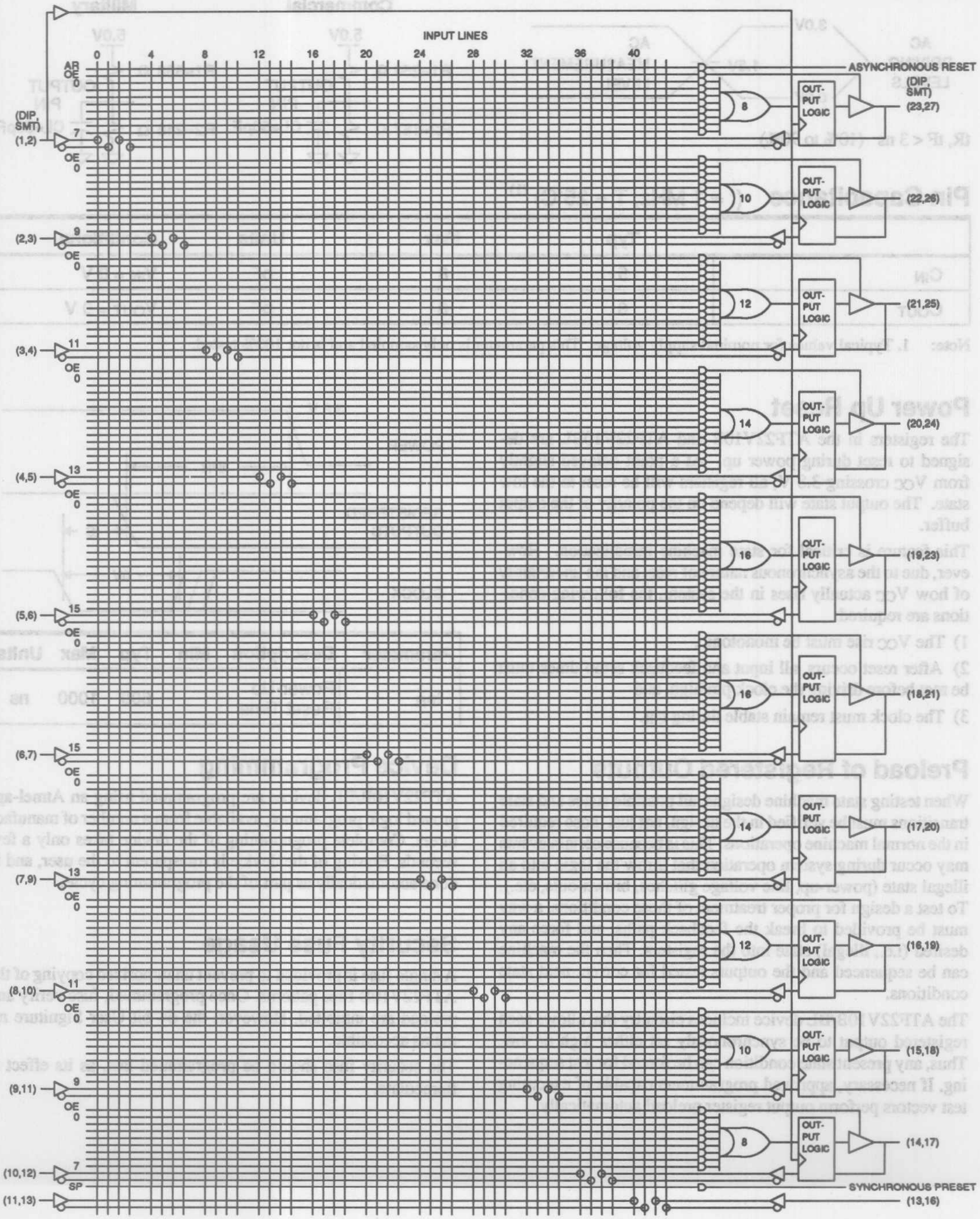
ATF22V10B/BL devices are programmed using an Atmel-approved logic programmer, available from a number of manufacturers. Complete programming of the device takes only a few seconds. Erasing of the device is transparent to the user, and is done automatically as part of the programming cycle.

Security Fuse Usage

A single fuse is provided to prevent unauthorized copying of the ATF22V10B fuse patterns. Once programmed, fuse verify and preload are inhibited. However, the 64 bit User Signiture remains accessible.

The security fuse should be programmed last, as its effect is immediate.

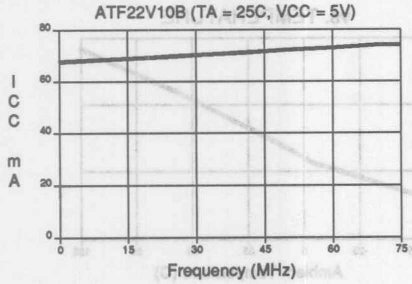
Functional Logic Diagram ATF22V10B/BL



1

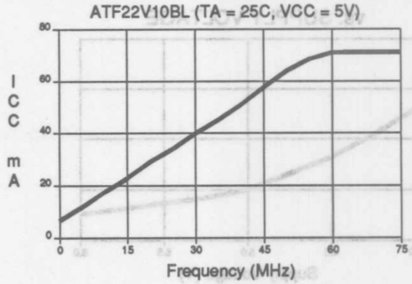
SUPPLY CURRENT vs. INPUT FREQUENCY

ATF22V10B (TA = 25C, VCC = 5V)

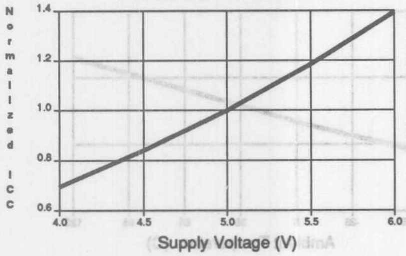


SUPPLY CURRENT vs. INPUT FREQUENCY

ATF22V10BL (TA = 25C, VCC = 5V)

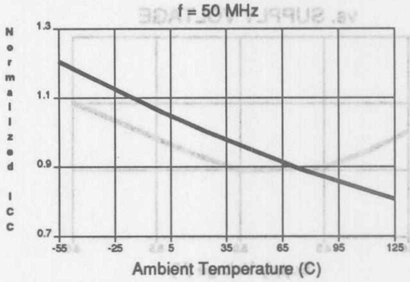


NORMALIZED SUPPLY CURRENT vs. SUPPLY VOLTAGE



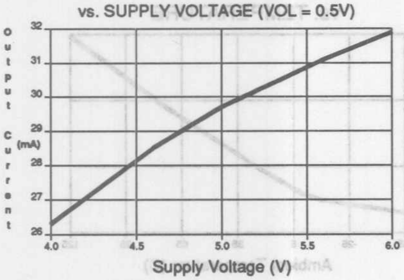
NORMALIZED ICC vs. AMBIENT TEMP.

f = 50 MHz



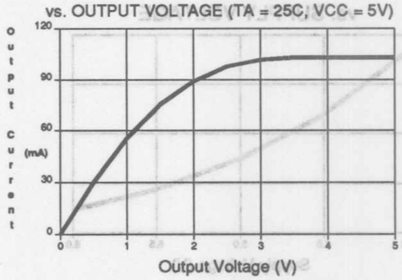
OUTPUT SINK CURRENT

vs. SUPPLY VOLTAGE (VOL = 0.5V)



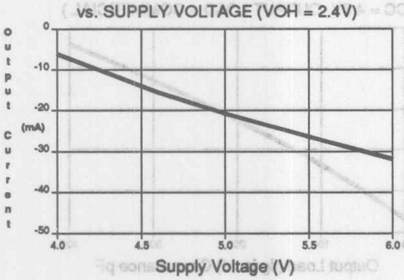
OUTPUT SINK CURRENT

vs. OUTPUT VOLTAGE (TA = 25C, VCC = 5V)



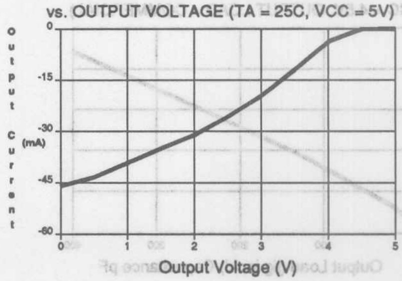
OUTPUT SOURCE CURRENT

vs. SUPPLY VOLTAGE (VOH = 2.4V)

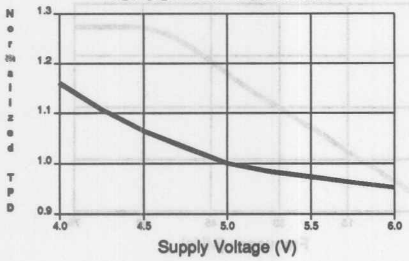


OUTPUT SOURCE CURRENT

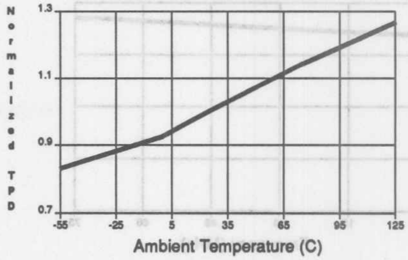
vs. OUTPUT VOLTAGE (TA = 25C, VCC = 5V)



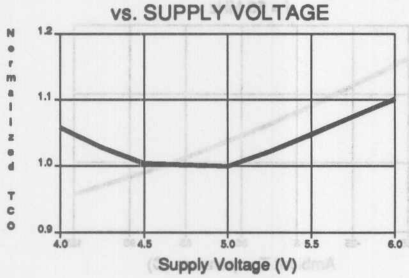
NORMALIZED TPD vs. SUPPLY VOLTAGE



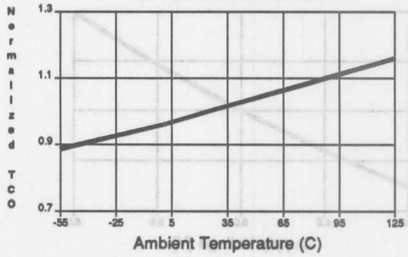
NORMALIZED TPD vs. TEMPERATURE



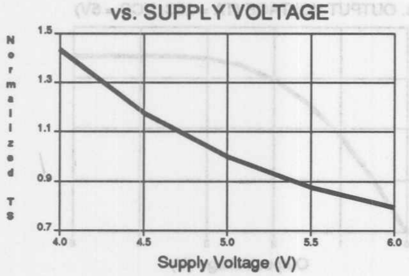
NORMALIZED TCO vs. SUPPLY VOLTAGE



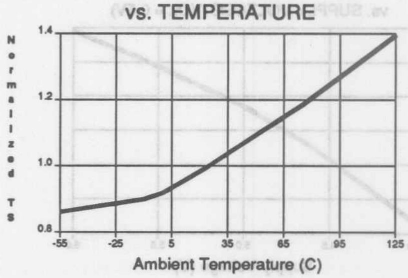
NORMALIZED TCO vs. TEMPERATURE



NORMALIZED TS vs. SUPPLY VOLTAGE

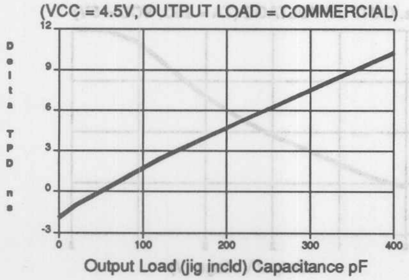


NORMALIZED TS vs. TEMPERATURE



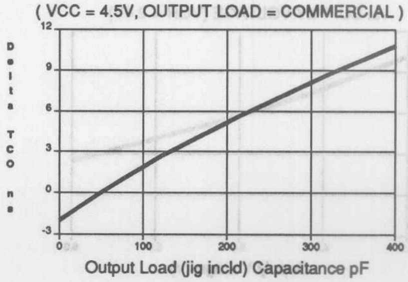
DELTA TPD vs. OUTPUT LOADING

(VCC = 4.5V, OUTPUT LOAD = COMMERCIAL)



DELTA TCO vs. OUTPUT LOADING

(VCC = 4.5V, OUTPUT LOAD = COMMERCIAL)



Ordering Information

t_{PD} (ns)	t_S (ns)	t_{CO} (ns)	Ordering Code	Package	Operation Range
7.5	6.5	5	ATF22V10B-7GC	24D3	Commercial (0°C to 70°C)
			ATF22V10B-7JC	28J	
			ATF22V10B-7PC	24P3	
10	7	7	ATF22V10B-10GC	24D3	Commercial (0°C to 70°C)
			ATF22V10B-10JC	28J	
			ATF22V10B-10PC	24P3	
			ATF22V10B-10SC	24S	
			ATF22V10B-10GI	24D3	Industrial (-40°C to 85°C)
			ATF22V10B-10JI	28J	
			ATF22V10B-10PI	24P3	
			ATF22V10B-10SI	24S	
			ATF22V10B-10GM	24D3	Military (-55°C to 125°C)
			ATF22V10B-10NM	28L	
			ATF22V10B-10GM/883	24D3	Military/883C (-55°C to 125°C) Class B, Fully Compliant
			ATF22V10B-10NM/883	28L	
15	10	8	ATF22V10B-15GC	24D3	Commercial (0°C to 70°C)
			ATF22V10B-15JC	28J	
			ATF22V10B-15PC	24P3	
			ATF22V10B-15SC	24S	
			ATF22V10B-15GI	24D3	Industrial (-40°C to 85°C)
			ATF22V10B-15JI	28J	
			ATF22V10B-15PI	24P3	
			ATF22V10B-15SI	24S	
			ATF22V10B-15GM	24D3	Military (-55°C to 125°C)
			ATF22V10B-15NM	28L	
			ATF22V10B-15GM/883	24D3	Military/883C (-55°C to 125°C) Class B, Fully Compliant
			ATF22V10B-15NM/883	28L	
25	15	15	ATF22V10B-25GC	24D3	Commercial (0°C to 70°C)
			ATF22V10B-25JC	28J	
			ATF22V10B-25PC	24P3	
			ATF22V10B-25SC	24S	
			ATF22V10B-25GI	24D3	Industrial (-40°C to 85°C)
			ATF22V10B-25JI	28J	
			ATF22V10B-25PI	24P3	
			ATF22V10B-25SI	24S	

Ordering Information

tPD (ns)	ts (ns)	tCO (ns)	Ordering Code	Package	Operation Range
10	7	7	ATF22V10BL-10GC ATF22V10BL-10JC ATF22V10BL-10PC	24D3 28J 24P3	Commercial (0°C to 70°C)
15	10	8	ATF22V10BL-15GC ATF22V10BL-15JC ATF22V10BL-15PC ATF22V10BL-15SC	24D3 28J 24P3 24S	Commercial (0°C to 70°C)
			ATF22V10BL-15GI ATF22V10BL-15JI ATF22V10BL-15PI ATF22V10BL-15SI	24D3 28J 24P3 24S	Industrial (-40°C to 85°C)
			ATF22V10BL-15GM ATF22V10BL-15NM	24D3 28L	Military (-55°C to 125°C)
			ATF22V10BL-15GM/883 ATF22V10BL-15NM/883	24D3 28L	Military/883C (-55°C to 125°C) Class B, Fully Compliant

Package Type	
24D3	24 Lead, 0.300" Wide, Ceramic Dual Inline Package (Cerdip)
28J	28 Lead, Plastic J-Leaded Chip Carrier (PLCC)
28L	28 Pad, Ceramic Leadless Chip Carrier (LCC)
24P3	24 Lead, 0.300" Wide, Plastic Dual Inline Package (PDIP)
24S	24 Lead, 0.300" Wide, Plastic Gull Wing Small Outline (SOIC)

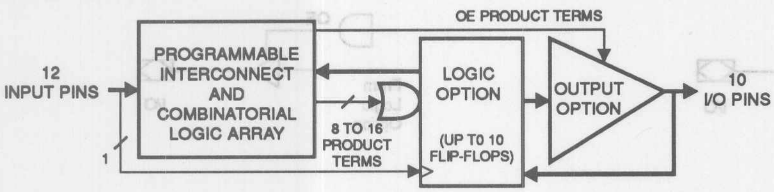
High Performance Flash PLD

Advanced Information

Features

- **Quarter Power Equivalent of ATF22V10B - 55 mA Maximum**
- **Low Power ATF22V10BQL - 10 mA Maximum Standby**
- **Industry Standard Architecture**
  Low-Cost, Easy-To-Use Software Tools
- **High Speed Electrically Erasable Programmable Logic Device**
  10 ns Max Propagation Delay
- **CMOS and TTL Compatible Inputs and Outputs**
  Input and I/O Pull-Up Resistors
- **Advanced Flash Technology**
  Reprogrammable
  100% Tested
- **High Reliability CMOS Technology**
  20 Year Data Retention
  100 Erase/Write Cycles
  2,000 V ESD Protection
  200 mA Latchup Immunity
- **Full Military, Commercial and Industrial Temperature Ranges**
- **Dual-In-Line and Surface Mount Packages in Standard Pinouts**

Logic Diagram



Description

The ATF22V10BQs are high performance CMOS (electrically erasable) Programmable Logic Devices (PLDs) that utilize Atmel's proven electrically erasable Flash memory technology. Speeds down to 10 ns and power dissipation as low as 10 mA are offered. All speed ranges are specified over the full 5 V ± 10% range for military and industrial temperature ranges, and 5 V ± 5% for commercial ranges.

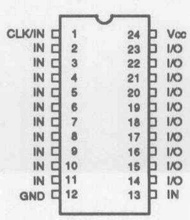
The ATF22V10BQL provides the fastest low power CMOS PLD solution, with low DC power (5.0 mA typical). The ATF22V10BQL significantly reduces total system power and enhances system reliability.

(continued)

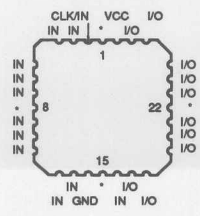
Pin Configurations

Pin Name	Function
CLK	Clock
IN	Logic Inputs
I/O	Bidirectional Buffers
*	No Internal Connection
VCC	+5 V Supply

DIP/SOIC



LCC/PLCC



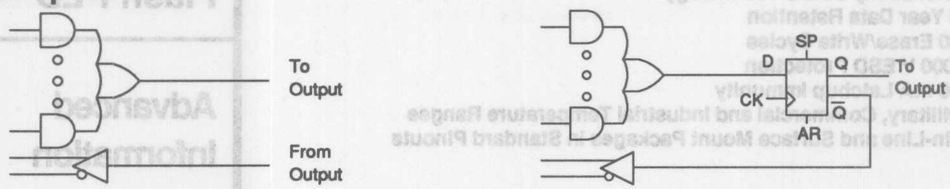
Description (Continued)

The ATF22V10BQs incorporate a variable product term architecture. Each output is allocated from eight to 16 product terms, which allows highly complex logic functions to be realized.

Two additional product terms are included to provide synchronous preset and asynchronous reset. These terms are common to all 10 registers. All registers are automatically cleared upon power up.

Register Preload simplifies testing. A Security Fuse prevents unauthorized copying of programmed fuse patterns.

Logic Options



Output Options

D.C. and A.C. Operating Conditions

	Commercial	Industrial	Military
Operating Temperature (Case)	0^oC - 70^oC	-40^oC - 85^oC	-55^oC - 125^oC
V_{CC} Power Supply	5 V ± 5%	5 V ± 10%	5 V ± 10%

Zero-Standby Power 20-Pin EPLD

1

Features

- 20-pin Universal EPLD
- Virtually Zero Standby Power
- Functional Replacement for Common 20-Pin Programmable Devices
  I_{OL} = 24 mA
- High Performance CMOS EPROM Cell Technology
  Erasable
  Reconfigurable
  100% Testable
- 25 ns and 35 ns Max Propagation Delay (Commercial)
- 30 ns and 40 ns Max Propagation Delay (Industrial)
- Up to 18 Inputs and 8 Input/Output Macrocells
- Programmable Output Polarity
- Power-Up Reset on all Registers
- Register Preload Capability
- Synchronous Preset/Asynchronous Reset
- Security Fuse to Protect Duplication of Proprietary Designs
- Design Support Provided using many Popular Software Development Packages for PLDs
- Available in 300-mil-wide DIP with Quartz Window, Plastic DIP (OTP), or PLCC (OTP)
- Second Source to Signetic's PLC18V8Z/I

Description

The AT18V8Z is a universal EPLD featuring high performance and virtually zero-standby power for power-sensitive applications. It is a reliable, user-configurable substitute for discrete TTL/CMOS logic. While compatible with TTL and HCT logic, the AT18V8Z can also replace HC logic over the V_{CC} range of 4.5 to 5.5 V.

The AT18V8Z is a two-level logic element comprised of ten inputs, 74 AND gates (product terms), and eight output Macrocells.

Each output features an "Output Macrocell" which can be individually configured as a dedicated input, a combinatorial output, or a registered output with internal feedback. As a result, the AT18V8Z is capable of emulating all common 20-pin programmable logic devices to reduce documentation, inventory, and manufacturing costs.

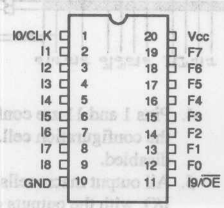
A power-up reset function and a Register Preload function have been incorporated into the AT18V8Z architecture to facilitate state machine design and testing.

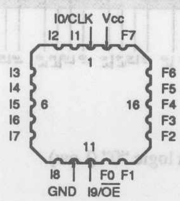
With a standby current of less than 100 µA and active power consumption of 1.5 mA/MHz, the AT18V8Z is ideally suited for power-sensitive applications in battery-operated/backed portable instruments and computers.

The AT18V8Z is also processed to industrial requirements for operation over an extended temperature range of -40°C to +85°C and supply voltage of 4.5 V to 5.5 V.

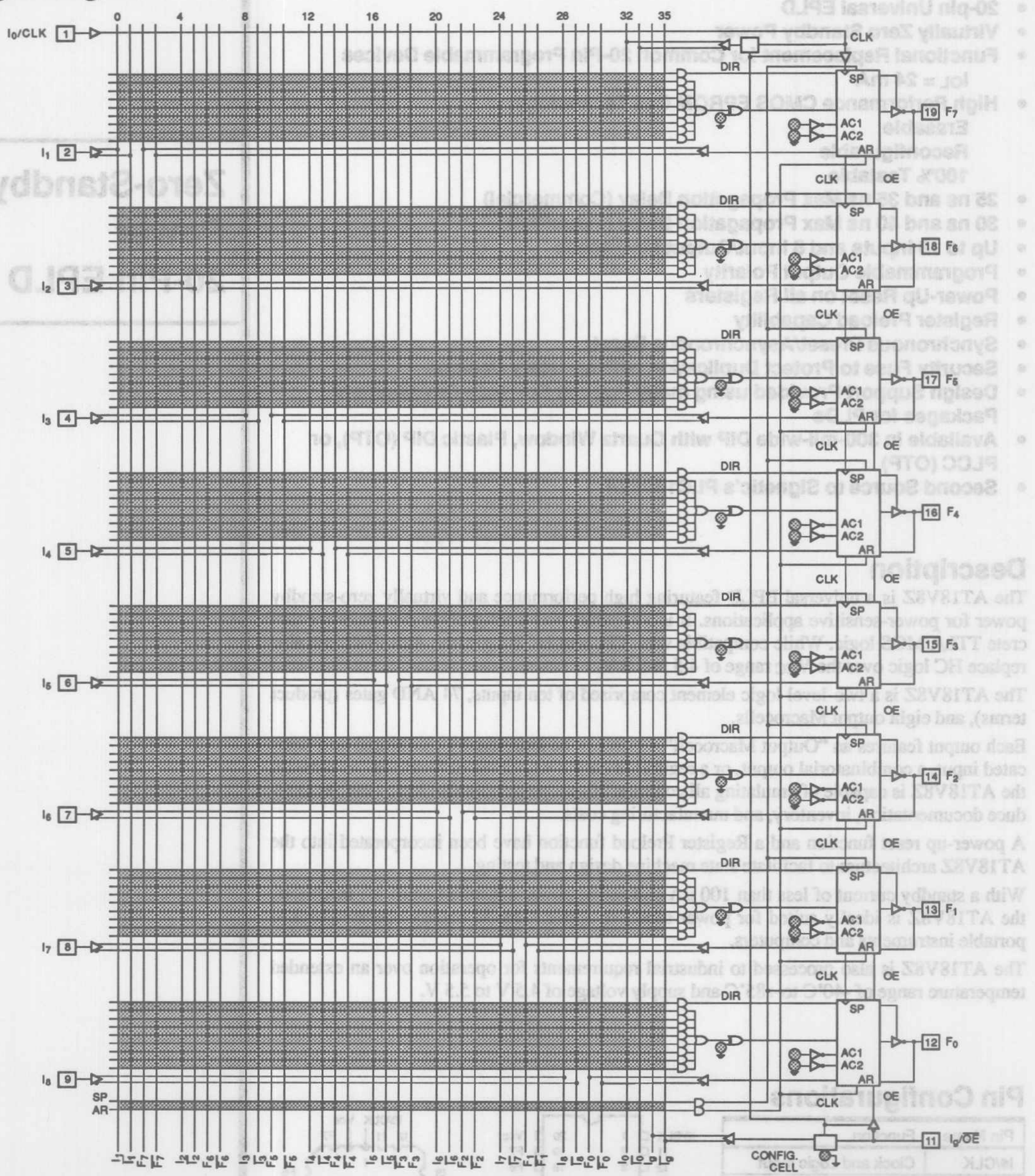
Pin Configurations

Pin Name	Function
I#/CLK	Clock and Logic Input
I#/OE	Output Enable and Logic Input
I#	Logic Inputs
F#	Bidirectional Buffers
*	No Internal Connection
V_{CC}	+5 V Supply





Logic Diagram



Notes:

In the unprogrammed or virgin state:

1. All cells are in a conductive state.
2. All AND gate locations are pulled to a logic "0" (Low).
3. Output polarity is inverting.
4. Pins 1 and 11 are configured as Inputs 0 and 9, respectively, via the configuration cell. The clock and $\overline{OE}$ functions are disabled.
5. All output macrocells (OMCs) are configured as bidirectional I/O, with the outputs disabled via the direction term.

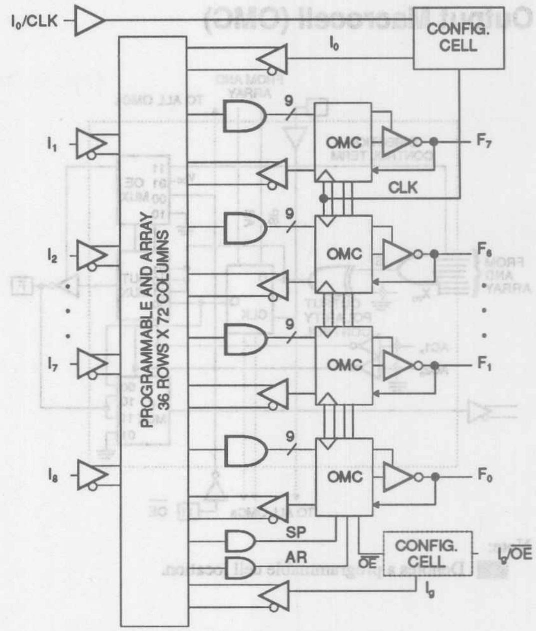
Denotes a programmable cell location.

Absolute Maximum Ratings*

Operating Temperature (Commercial)	0°C to +75°C
Operating Temperature (Industrial)	-40°C to +85°C
Storage Temperature	-65°C to +150°C
Supply Voltage	-0.5 to +7 VDC
Operating Supply Voltage (Commercial)	4.75 to 5.25 VDC
Operating Supply Voltage (Industrial)	4.5 to 5.5 VDC
Input Voltages (including N.C. Pins) with Respect to Ground	-0.5 V to VCC+0.5 VDC
Output Voltages with Respect to Ground	-0.5 V to VCC+0.5 VDC
Input Currents	-10 to +10 mA
Output Currents	+24 mA

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Functional Diagram



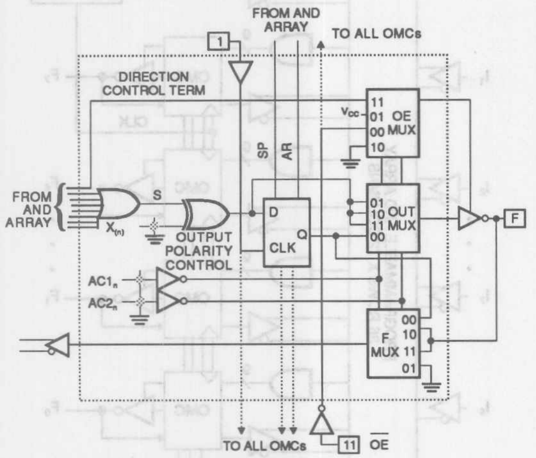
PAL Device to AT18V8Z Output Pin Configuration Cross Reference

Pin	AT18V8Z	16L8 16H8 16P8 16P8	16R4 16RP4	16R6 16RP6	16R8 16RP8	16L2 16H2 16P2	14L4 14H4 14P4	12L6 12H6 12P6	10L8 10H8 10P8
1	I_0/CLK	I	CLK	CLK	CLK	I	I	I	I
19	F7	B	B	B	D	I	I	I	O
18	F6	B	B	D	D	I	I	O	O
17	F5	B	D	D	D	I	O	O	O
16	F4	B	D	D	D	O	O	O	O
15	F3	B	D	D	D	O	O	O	O
14	F2	B	D	D	D	I	O	O	O
13	F1	B	B	D	D	I	I	O	O
12	F0	B	B	B	D	I	I	I	O
11	I_9/$\overline{OE}$	I	$\overline{OE}$	$\overline{OE}$	$\overline{OE}$	I	I	I	I

The Atmel state-of-the-art floating-gate CMOS EPROM process yields bipolar equivalent performance at less than one-quarter the power consumption. The erasable nature of the EPROM process enables Atmel to functionally test the devices prior to shipment to the customer. Additionally, this allows Atmel to extensively stress test, as well as ensure the threshold voltage of each individual EPROM cell. 100% programming yield is subsequently guaranteed.

PAL may be a registered trademark of AMD Corp.

Output Macrocell (OMC)



Note:
▒ Denotes a programmable cell location.

The Output Macrocell (OMC)

The AT18V8Z series devices have eight individually programmable Output Macrocells. The 72 AND inputs (or product terms) from the programmable AND array are connected to the eight OMCs in groups of nine. Eight of the AND terms are dedicated to logic functions; the ninth is for asynchronous direction control, which enables/disables the respective bidirectional I/O pin. Two product terms are dedicated for the Synchronous Preset and Asynchronous Reset functions.

Each OMC can be independently programmed via 16 architecture control bits, $AC1_n$ and $AC2_n$ (one pair per macrocell). Similarly, each OMC has a programmable output polarity control bit (X_n). By configuring the pair of architecture control bits according to the configuration cell table, four different configurations may be implemented. Note that the configuration cell is automatically programmed based on the OMC configuration.

Design Security

The AT18V8Z series devices have a programmable security fuse that controls the access to the data programmed in the device.

Configuration Cell

A single configuration cell controls the functions of Pins 1 and 11. Refer to Functional Diagram. When the configuration cell is programmed, Pin 1 is a dedicated clock and Pin 11 is dedicated for output enable. When the configuration cell is unprogrammed, Pins 1 and 11 are both dedicated inputs. Note that the output enable for all registered OMCs is common—from Pin 11 only. Output enable control of the bidirectional I/O OMCs is provided from the AND array via the direction product term.

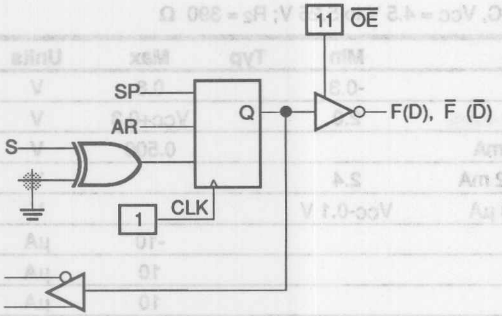
If any one OMC is configured as registered, the configuration cell will be automatically configured (via the design software) to ensure that the clock and output enable functions are enabled on Pins 1 and 11, respectively. If none of the OMCs are registered, the configuration cell will be programmed such that Pins 1 and 11 are dedicated inputs. The programming codes are as follows:

Pin 1 = CLK, Pin 11 = $\overline{OE}$	L
Pin 1 and Pin 11 = Input	H

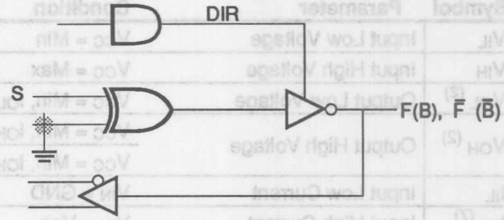
Function	Control Cell Configurations			Comments
	$AC1_n$	$AC2_n$	Config. Cell	
Registered Mode	Programmed	Programmed	Programmed	Dedicated clock from Pin 1. $\overline{OE}$ control for all registered OMCs from Pin 11 only.
Bidirectional I/O mode(1)	Unprogrammed	Unprogrammed	Unprogrammed	Pins 1 and 11 are dedicated inputs. 3-State control from AND array only.
Fixed input mode	Unprogrammed	Programmed	Unprogrammed	Pins 1 and 11 are dedicated inputs.
Fixed output mode	Programmed	Unprogrammed	Unprogrammed	Pins 1 and 11 are dedicated inputs. The feedback path (via F_{MUX}) is disabled.

Note:
1. This is the virgin state as shipped by the factory.

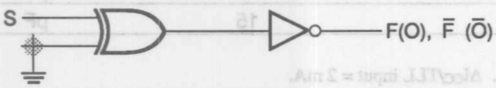
Architecture Control: AC1 and AC2



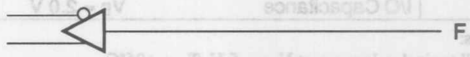
OMC Configuration	Code
Registered (D-type)	D



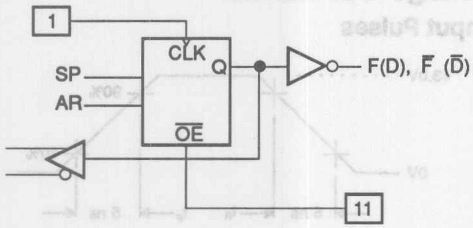
OMC Configuration	Code
Bidirectional I/O[1] (Combinatorial)	B



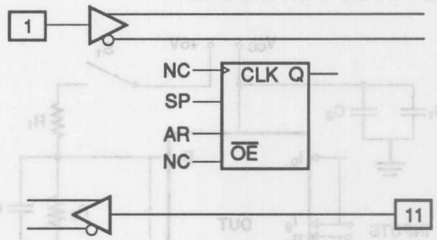
OMC Configuration	Code
Fixed Output	O



OMC Configuration	Code
Fixed Input	I



OMC Configuration	Code
Pin 1 = CLK Pin 11 = $\overline{OE}$	L



OMC Configuration	Code
Pin 1 = Input Pin 11 = Input	H[6]

Notes:

A factory shipped unprogrammed device is configured such that:

1. This is the initial unprogrammed state. All cells are in a conductive state.
2. All AND gates are pulled to a logic "O" (Low).
3. Output polarity is inverting.
4. Pins 1 and 11 are configured as inputs 0 and 9. The clock and $\overline{OE}$ functions are disabled.
5. All Output Macrocells (OMCs) are configured as bidirectional I/O, with the outputs disabled via the direction term.
6. This configuration cannot be used if any OMCs are configured as registered (Code = D). The configuration cell will be automatically configured to ensure that the clock and output enable functions are enabled on Pins 1 and 11, respectively, if any one OMC is programmed as registered.

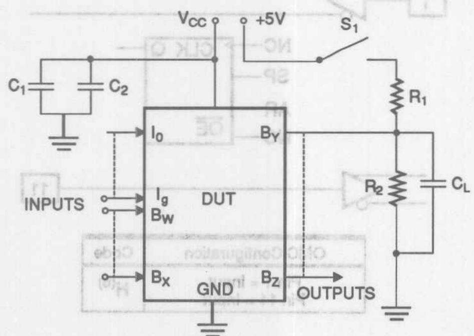
D.C. Characteristics

T_{AC} = 0°C to +75°C, V_{CC} = 4.75 V to 5.25 V; T_{AI} = -40°C to 85°C, V_{CC} = 4.5 V to 5.25 V; R_2 = 390 Ω

Symbol	Parameter	Condition	Min	Typ	Max	Units
V_{IL}	Input Low Voltage	V_{CC} = Min	-0.3		0.8	V
V_{IH}	Input High Voltage	V_{CC} = Max	2.0		V_{CC}+0.3	V
V_{OL} (2)	Output Low Voltage	V_{CC} = Min, I_{OL} = 24 mA			0.500	V
V_{OH} (2)	Output High Voltage	V_{CC} = Min, I_{OH} = -3.2 mA	2.4			V
		V_{CC} = Min, I_{OH} = -20 µA	V_{CC}-0.1 V			V
I_{IL}	Input Low Current	V_{IN} = GND			-10	µA
I_{IH} (7)	Input High Current	V_{IN} = V_{CC}			10	µA
$I_{O(OFF)}$	Hi-Z State Output Current	V_{OUT} = V_{CC}			10	µA
		V_{OUT} = GND			-10	µA
I_{OS} (3)	Short Circuit Output Current	V_{OUT} = GND			-130	mA
I_{CC}	V_{CC} Standby Supply Current	V_{CC} = Max, V_{IN} = 0 or V_{CC} (8)			100	µA
$I_{CC/f}$ (4)	V_{CC} Active Supply Current	V_{CC} = Max (CMOS Inputs) (5,6)			1.5	mA/MHz
C_I	Input Capacitance	V_{CC} = 5 V, V_{IN} = 2.0 V		12		pF
C_B	I/O Capacitance	V_B = 2.0 V		15		pF

Notes:
1. All typical values are at V_{CC} = 5 V, T_A = +25°C.
2. All voltage values are with respect to network ground terminal.
3. Duration of short-circuit should not exceed one second. Test one at a time.
4. Tested with TLL input levels: V_{IL} = 0.45 V, V_{IH} = 2.4 V. Measured with all outputs switching.
5. ΔI_{CC}/TLL input = 2 mA.
6. ΔI_{CC} vs. frequency (registered configuration) = 2 mA/MHz.
7. I_{IL} for Pin 1 (I_O/CLK) is ± 10 µA with V_{IN} = 0.4 V.
8. V_{IN} includes CLK and $\overline{OE}$ if applicable.

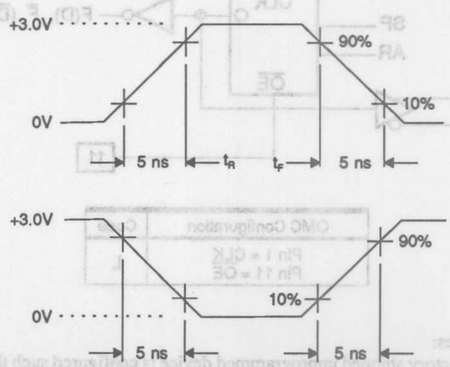
A.C. Test Conditions



Notes:
C_1 and C2 are to bypass V_{CC} to GND.
C_L = 50 pF
R_1 = 200 Ω
R_2 = 390 Ω

Voltage Waveforms

Input Pulses



Note: All circuit delays are measured at the +1.5 V level of inputs and outputs, unless otherwise specified.

A.C. Read Characteristics

$T_{AC} = 0°C$ to $+75°C$, $V_{CC} = 4.75$ V to 5.25 V; $T_{AI} = -40°C$ to 85°C, $V_{CC} = 4.5$ V to 5.25 V; $R_2 = 390\ \Omega$

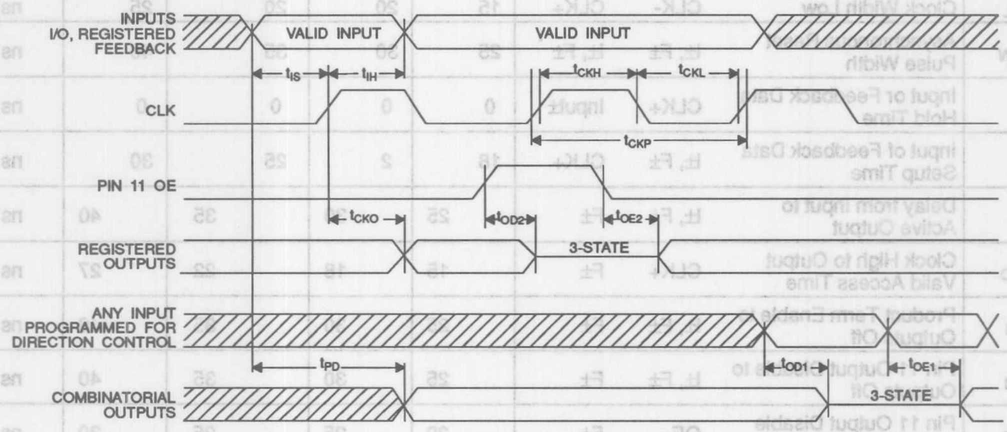
				AT18V8Z								
				-25		-30		-35 (Comm.)		-35 (Ind.)		
Symbol	Parameter	From	To	Min	Max	Min	Max	Min	Max	Min	Max	Units
t_{CKP}	Clock Period (Minimum $t_{IS} + t_{CKO}$)	CLK+	CLK+	33		40		47		57		ns
t_{CKH}	Clock Width High	CLK+	CLK-	15		20		20		25		ns
t_{CKL}	Clock Width Low	CLK-	CLK+	15		20		20		25		ns
t_{ARW}	Asynchronous Reset Pulse Width	I±, F±	I±, F±	25		30		35		40		ns
t_{IH}	Input or Feedback Data Hold Time	CLK+	Input±	0		0		0		0		ns
t_{IS}	Input of Feedback Data Setup Time	I±, F±	CLK+	18		2		25		30		ns
t_{PD}	Delay from Input to Active Output	I±, F±	F±		25		30		35		40	ns
t_{CKO}	Clock High to Output Valid Access Time	CLK+	F±		15		18		22		27	ns
t_{OE1}	Product Term Enable to Outputs Off	I±, F±	F±		25		30		35		40	ns
t_{OD1}	Pin 11 Output Disable to Outputs Off	I±, F±	F±		25		30		35		40	ns
t_{OD2}	Pin 11 Output Disable High to Outputs Off	OE-	F±		20		25		25		30	ns
t_{OE2}	Pin 11 Output Enable to Active Output	OE+	F±		20		25		25		30	ns
t_{ARD}	Asynchronous Reset Delay	I±, F±	F+		30		35		35		40	ns
t_{ARR}	Asynchronous Reset Recovery Time	I±, F±	CLK+	20		25		25		30		ns
t_{SPR}	Synchronous Preset Recovery Time	I±, F±	CLK+	20		25		25		30		ns
t_{PPR}	Power-Up Reset	V_{CC}+	F+		25		30		35		40	ns
f_{MAX}	Maximum Frequency	$I/(t_{IS} + t_{CKO})$			30		25		21		18	ns

Power Up Reset

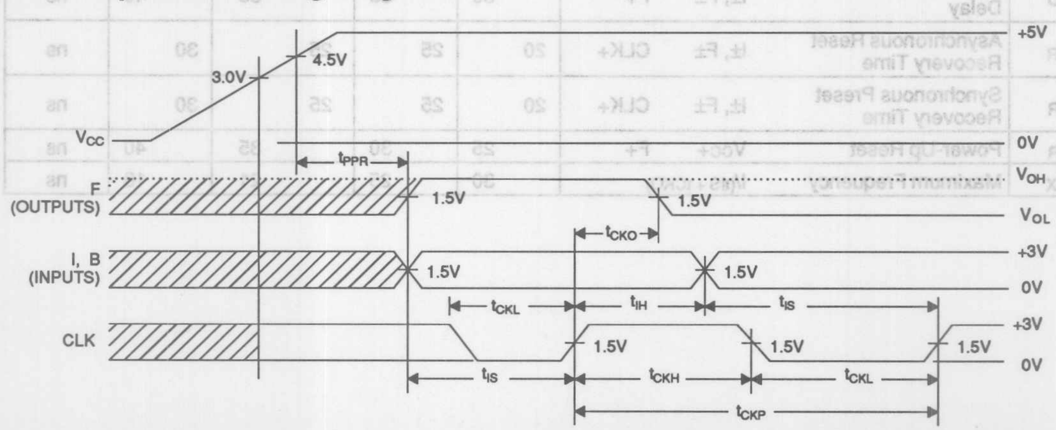
In order to facilitate state machine design and testing, a power-up reset function has been incorporated in the AT18V8Z. All internal registers will reset to Active-Low (logical "0") after a specified period of time (t_{PPR}).

Therefore, any OMC that has been configured as a registered output will always produce an Active-High on the associated output pin because of the inverted output buffer. The internal feedback (Q) of a registered OMC will also be set Low. The programmed polarity of OMC will not affect the Active-High output condition during a system power-up condition.

Switching Waveforms Timing Diagram

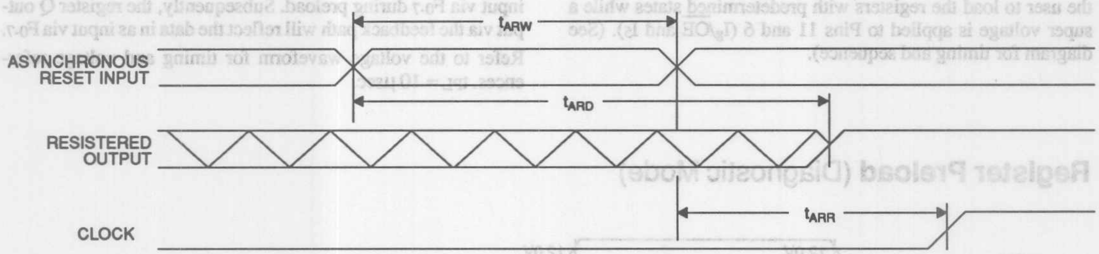


Power Up Reset Timing Diagram

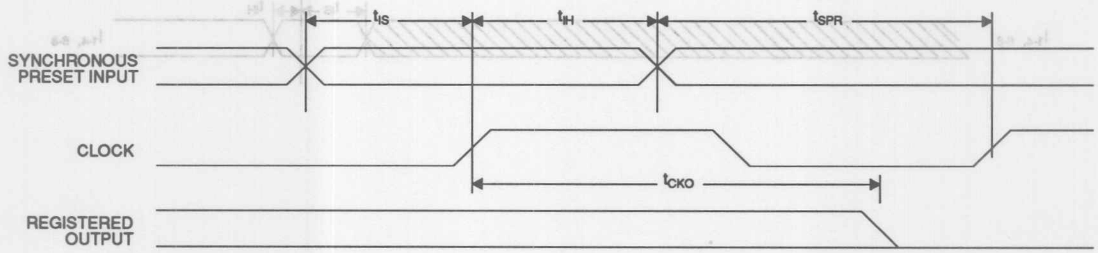


Note:
Diagram presupposes that the outputs (F) are enabled. The reset occurs regardless of the output condition (enabled or disabled).

Asynchronous Reset Timing Diagram



Synchronous Preset Timing Diagram



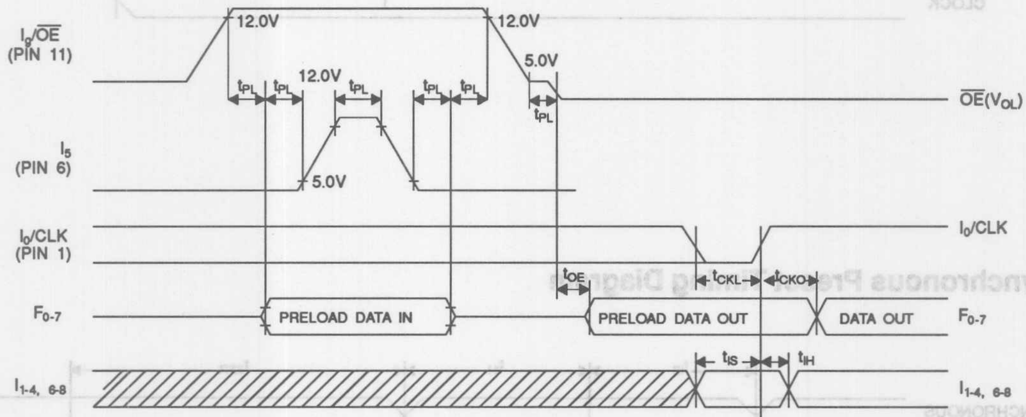
Register Preload Function (Diagnostic Mode Only)

In order to facilitate the testing of state machine/controller designs, a diagnostic mode register preload feature has been incorporated into the AT18V8Z series device. This feature enables the user to load the registers with predetermined states while a super voltage is applied to Pins 11 and 6 ($I_8/\overline{OE}$ and I_5). (See diagram for timing and sequence).

To read the data out, Pins 11 and 6 must be returned to normal TLL levels. The outputs, $F_{0\text{-}7}$, must be enabled in order to read data out. The Q outputs of the registers will reflect data in as input via $F_{0\text{-}7}$ during preload. Subsequently, the register Q output via the feedback path will reflect the data in as input via $F_{0\text{-}7}$.

Refer to the voltage waveform for timing and voltage references. t_{PL} = 10 µsec.

Register Preload (Diagnostic Mode)



Logic Programming

The AT18V8Z series is fully supported by industry standard (JEDEC compatible) PLD CAD tools. ABEL™ and CUPL™ design packages also support the AT18V8Z architecture.

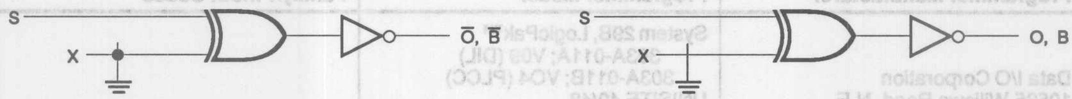
All packages allow Boolean and state equation entry formats. ABEL and CUPL also accept, as input, schematic capture format.

AT18V8Z logic designs can also be generated using the program table entry format, which is detailed on the following pages.

With Logic programming, the AND/OR/Ex-OR gate input connections necessary to implement the desired logic function are coded directly from logic equations using the Program Table. Similarly, various OMC configurations are implemented by programming the Architecture Control bits AC1 and AC2. Note that the configuration cell is automatically programmed based on the OMC configuration.

In this table, the logic state of variables I, P, and B associated with each Sum Term S is assigned a symbol which results in the proper fusing pattern of corresponding link pairs, defined as follows:

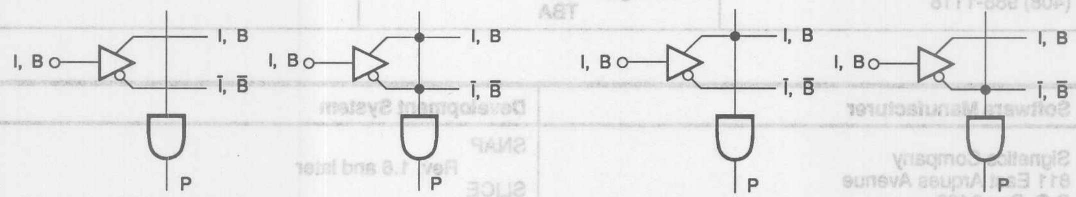
Output Polarity (O, B)



Active Level	Code
Inverting[1]	L

Active Level	Code
Non-inverting	H

"AND" Array (I, B)



State	Code
Don't Care	–

State	Code
Inactive[1]	O

State	Code
I, B	H

State	Code
$\overline{I}, \overline{B}$	L

Note:
1. A factory-shipped unprogrammed device is configured such that all cells are in a conductive state.

ABEL and CUPL may be registered trademarks of others.

Erasure Characteristics (For Quartz Window Packages Only)

The erasure characteristics of the AT18V8Z Series devices are such that erasure begins to occur upon exposure to light with wavelengths shorter than approximately 4000 Angstroms (Å). It should be noted that sunlight and certain types of fluorescent lighting could erase a typical AT18V8Z in approximately three years, while it would take approximately one week to cause erasure when exposed to direct sunlight. If the AT18V8Z is to be exposed to these types of lighting conditions for extended periods of time, opaque labels should be placed over the window to prevent unintentional erasure.

The recommended erasure procedure for the AT18V8Z is exposure to shortwave ultraviolet light which has a wavelength of 2537 Angstroms (Å). The integrated dose (i.e., UV intensity x exposure time) for erasure should be a minimum of 15 Wsec/cm^2. The erasure time with this dosage is approximately 30 to 35 minutes using an ultraviolet lamp with a 12,000 µW/cm^2 power rating. The device should be placed within one inch of the lamp tubes during erasure. The maximum integrated dose a CMOS EPLD can be exposed to without damage is 7258 Wsec/cm^2. Exposure of these CMOS EPLDs to high intensity UV light for longer periods may cause permanent damage.

Programing

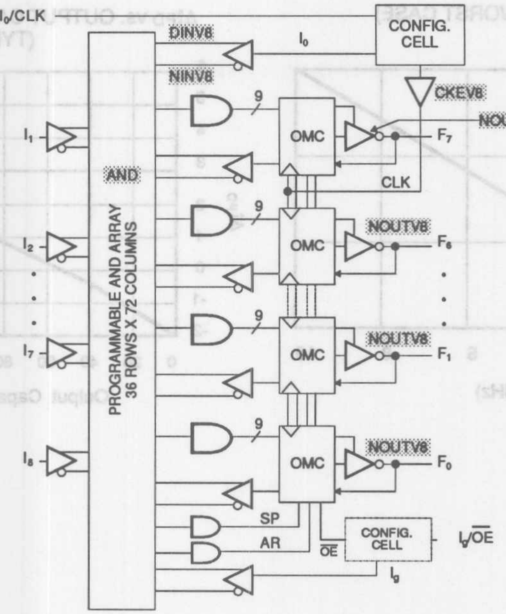
The AT18V8Z is programmable on conventional programmers for 20-pin PAL devices.

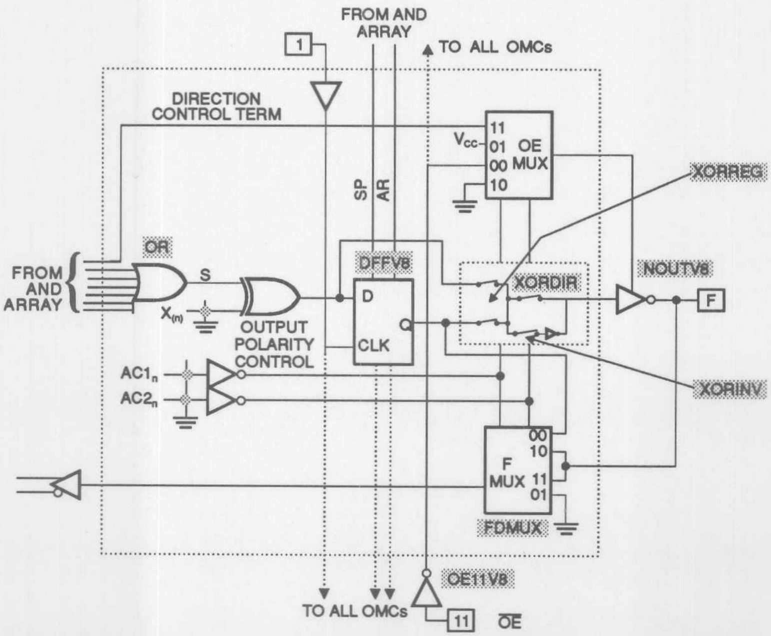
Refer to the following charts for qualified manufacturers of programmers and software tools:

Programmer Manufacturer	Programmer Model	Family/Pinout Codes
Data I/O Corporation 10525 Willows Road, N.E. P.O. Box 97046 Redmond, Washington 98073-9746 (800) 247-5700	System 29B, LogicPak™ 303A-011A; V09 (DIL) 303A-011B; VO4 (PLCC) UNISITE 40/48 V2.6 (DIL) Chipsite (PLCC)-V2.8 MODEL 60 360A001 (DIL) 360A006 (PLCC)	86/4F
Stag Microsystems, Inc. 1600 Wyatt Drive, Suite 3 Santa Clara, California 95054 (408) 988-1118	ZL30/30A Programmer Rev. 30A34 (DIL) 30A001 Adaptor (PLCC) PPZ Programmer TBA	12/205

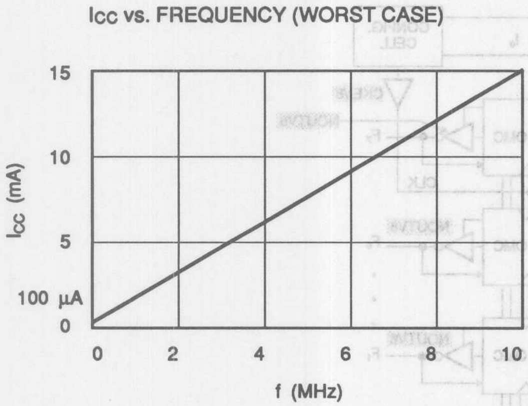
Software Manufacturer	Development System
Signetics Company 811 East Arques Avenue P.O. Box 3409 Sunnyvale, California 94088-3409 (408) 991-2000	SNAP Rev. 1.6 and later SLICE Rev. 1.0 and later AMAZE Software Rev. 1.8 and later
Data I/O Corporation 10525 Willows Road, N.E. P.O. Box 97046 Redmond, Washington 98073-9746 (800) 247-5700	ABEL™ Software
Logical Devices, Inc. 1201 Northwest 65th Place Fort Lauderdale, Florida 33309 (800) 331-7766	CUPL™ Software

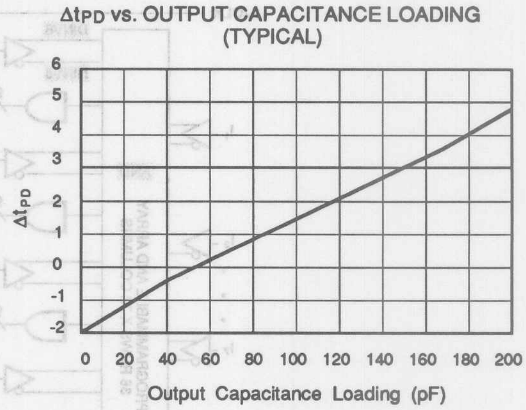
Snap Resource Summary Designations





Note:
▩ Denotes a programmable cell location.


Icc vs. FREQUENCY (WORST CASE)
15
10
5
100 µA
0
Icc (mA)
0
2
4
6
8
10
f (MHz)


ΔtPD vs. OUTPUT CAPACITANCE LOADING (TYPICAL)
6
5
4
3
2
1
0
-1
-2
ΔtPD
0
20
40
60
80
100
120
140
160
180
200
Output Capacitance Loading (pF)

Ordering Information

t_{PD} (ns)	t_S (ns)	t_{CO} (ns)	Ordering Code	Package	Operation Range
25	18	15	AT18V8Z-25DC AT18V8Z-25JC AT18V8Z-25PC	20DW3 20J 20P3	Commercial (0°C to 70°C)
30	22	18	AT18V8Z-30DI AT18V8Z-30JI AT18V8Z-30PI	20DW3 20J 20P3	Industrial (-40°C to 85°C)
35	25	22	AT18V8Z-35DC AT18V8Z-35JC AT18V8Z-35PC	20DW3 20J 20P3	Commercial (0°C to 70°C)
40	30	27	AT18V8Z-40DI AT18V8Z-40JI AT18V8Z-40PI	20DW3 20J 20P3	Industrial (-40°C to 85°C)

Package Type	
20DW3	20 Lead, 0.300" Wide, Windowed, Ceramic Dual Inline Package (Cerdip)
20J	20 Lead, Plastic J-Leaded Chip Carrier OTP (PLCC)
20P3	20 Lead, 0.300" Wide Plastic Dual Inline Package OTP (PDIP)

High Speed UV Erasable Programmable Logic Device

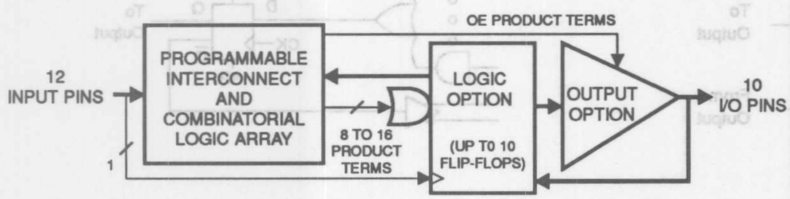
Features

- **High Speed Programmable Logic Device**
  15 ns Max Propagation Delay
  5 V ±10% Operation
- **Low Power CMOS Operation**

Speed	"L"	-15,-20	All
Temp	Com./Mil.	Com./Mil.	Others
Icc(mA)	12/15	90/100	55

- **CMOS and TTL Compatible Inputs and Outputs**
  10 µA Leakage Maximum
- **Reprogrammable - Tested 100% for Programmability**
- **High Reliability CMOS Technology**
  2000 V ESD Protection
  200 mA Latchup Immunity
- **Full Military, Commercial and Industrial Temperature Ranges**
- **Dual-In-Line and Surface Mount Packages**

Logic Diagram



Description

The AT22V10 and AT22V10L are CMOS high performance EPROM-based Programmable Logic Devices (PLDs). Speeds down to 15 ns and power dissipation as low as 12 mA are offered. All speed ranges are specified over the full 5 V ±10% range. All pins offer a low ±10 µA leakage.

The AT22V10L provides the optimum low power CMOS PLD solution, with low DC power (8 mA typical) and full CMOS output levels. The AT22V10L significantly reduces total system power and enhances system reliability.

Full CMOS output levels help reduce power in many other system components.

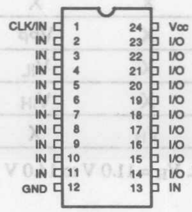
The AT22V10 and AT22V10L incorporate a variable product term architecture. Each output is allocated from eight to 16 product terms, which allows highly complex logic functions to be realized.

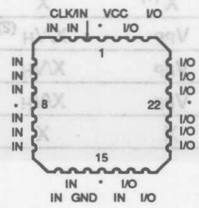
Two additional product terms are included to provide synchronous preset and asynchronous reset. These terms are common to all 10 registers. All registers are automatically cleared upon power up.

Register preload simplifies testing. A security fuse prevents unauthorized copying of programmed fuse patterns.

Pin Configurations

Pin Name	Function
CLK/IN	Clock and Logic Input
IN	Logic Inputs
I/O	Bidirectional Buffers
*	No Internal Connection
VCC	+5 V Supply





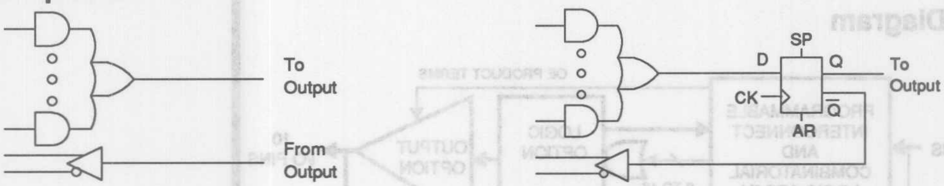
Absolute Maximum Ratings*

Temperature Under Bias	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-2.0 V to +7.0 V(1)
Voltage on Input Pins with Respect to Ground During Programming	-2.0 V to +14.0 V(1)
Programming Voltage with Respect to Ground	-2.0 V to +14.0 V(1)
Integrated UV Erase Dose	7258 W·sec/cm^2

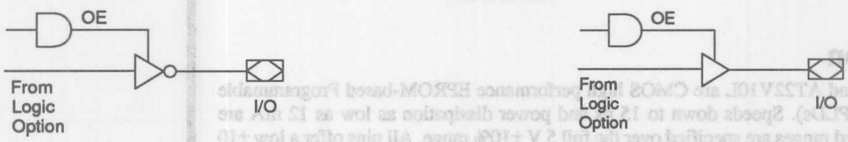
*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Note:
1. Minimum voltage is -0.6 V dc which may undershoot to -2.0 V for pulses of less than 20 ns. Maximum output pin voltage is V_{CC}+0.75 V dc which may overshoot to +7.0 V for pulses of less than 20 ns.

Logic Options



Output Options



D.C. and A.C. Operating Conditions

	Commercial AT22V10/L -15, -20, -25	Industrial AT22V10/L -15, -20, -25	Military AT22V10/L -15, -20, -25, -30
Operating Temperature (Case)	0°C - 70°C	-40°C - 85°C	-55°C - 125°C
V_{CC} Power Supply	5 V ± 10%	5 V ± 10%	5 V ± 10%

Operating Modes

Mode	24-Pin DIP 28-Pin JLCC	1 2	5 6	8 10	13 16	I/Os I/Os	V_{CC}(24) V_{CC}(28)
"PLD"		X (1)	X	X	X	I/O	5 V
Program		V_{PP}	X/V_H (2)	X	V_{PP}	D_{IN}	6 V
PGM Verify		V_{PP}	X/V_H	X	V_{IL}	D_{OUT}	6 V
PGM Inhibit		V_{PP}	X/V_H	X	V_{IH}	High Z	6 V
Preload		X	X	V_H	X	D_{IN}	5 V

Notes: 1. X can be V_{IL} or V_{IH}. 2. V_H = 11.0 V to 14.0 V

D.C. Characteristics

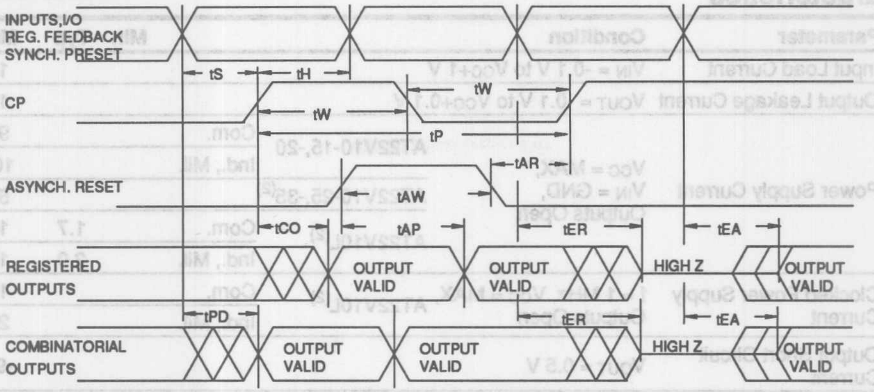
Symbol	Parameter	Condition			Min	Typ	Max	Units
I_{LI}	Input Load Current	V_{IN} = -0.1 V to V_{CC}+1 V					10	µA
I_{LO}	Output Leakage Current	V_{OUT} = -0.1 V to V_{CC}+0.1 V					10	µA
I_{CC}	Power Supply Current	V_{CC} = MAX, V_{IN} = GND, Outputs Open	AT22V10-15,-20	Com.			90	mA
				Ind., Mil.			100	mA
			AT22V10-25,-35[2]				55	mA
			AT22V10L[2]	Com.		1.7	12	mA
				Ind., Mil.		2.0	15	mA
I_{CC2}	Clocked Power Supply Current	f = 1 MHz, V_{CC} = MAX, Outputs Open	AT22V10L[2]	Com.			15	mA
				Ind., Mil.			20	mA
I_{OS}[1]	Output Short Circuit Current	V_{OUT} = 0.5 V					-90	mA
V_{IL}	Input Low Voltage				-0.6		0.8	V
V_{IH}	Input High Voltage				2.0		V_{CC}+0.75	V
V_{OL}	Output Low Voltage	V_{IN} = V_{IH} or V_{IL}, V_{CC} = MIN	I_{OL} = 16 mA	Com.,Ind.			0.5	V
			I_{OL} = 12 mA	Mil.			0.5	V
			I_{OL} = 24 mA	Com.			0.8	V
V_{OH}	Output High Voltage	V_{IN}=V_{IH} or V_{IL}, V_{CC}=MIN	I_{OH} = -100 µA		V_{CC}-0.3			V
			I_{OH} = -4.0 mA		2.4			V

Notes: 1. Not more than one output at a time should be shorted. Duration of short circuit test should not exceed 30 sec.
2. See I_{CC} vs. Frequency curves in the back of this data sheet.

A.C. Characteristics, Commercial and Industrial

Symbol	Parameter	AT22V10-15			AT22V10/L-20			AT22V10/L-25			Units
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
t_{PD}	Input or Feedback to Non-Registered Output		10	15		12	20		15	25	ns
t_{EA}	Input to Output Enable		10	15			20		15	25	ns
t_{ER}	Input to Output Disable		10	15			20		15	25	ns
t_{CF}	Clock to Feedback	0	1	2.5	0	4	8	0	5	10	ns
t_{CO}	Clock to Output	0	7	10	0	8	12	0	10	15	ns
t_S	Input or Feedback Setup Time	10	8		12	8		15	12		ns
t_H	Hold Time	0			0			0			ns
t_P	Clock Period	12			20			24			ns
t_W	Clock Width	6			10			12			ns
F_{MAX}	External Feedback $1/(t_S+t_{CO})$			50.0			41.6			33.3	MHz
	Internal Feedback $1/(t_S + t_{CF})$			80.0			50.0			40.0	MHz
	No Feedback $1/(t_P)$			83.3			50.0			41.6	MHz
t_{AW}	Asynchronous Reset Width	15	8		20	9		25	10		ns
t_{AR}	Asynchronous Reset, Synchronous Preset, Recovery Time	15	8		20	12		25	15		ns
t_{AP}	Asynchronous Reset to Registered Output Reset		12	20		15	22		18	25	ns

A.C. Waveforms [1]

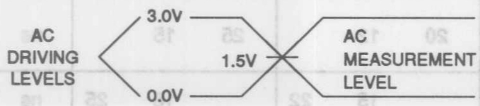


Note: 1. Timing measurement reference is 1.5 V. Input AC driving levels are 0.0 V and 3.0 V, unless otherwise specified.

A.C. Characteristics, Military

Symbol	Parameter	AT22V10-15			AT22V10/L-20			AT22V10/L-25			AT22V10/L-30			Units
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
t_{PD}	Input or Feedback to Non-Registered Output		10	15		12	20		15	25		20	30	ns
t_{EA}	Input to Output Enable		10	15			20		15	25		20	30	ns
t_{ER}	Input to Output Disable		10	15			20		15	25		20	30	ns
t_{CF}	Clock to Feedback	0	1	2.5	0	4	8	0	5	10	0	10	15	ns
t_{CO}	Clock to Output	0	7	10	0	8	15	0	10	15	0	12	20	ns
t_{SF}	Feedback Setup Time	10	8		12	10		15	12		18	15		ns
t_S	Input Setup Time	10	8		17	14		18	15		20	15		ns
t_H	Hold Time	0			0			0			0			ns
t_P	Clock Period	12			20			24			30			ns
t_W	Clock Width	6			10			12			15			ns
	External Feedback $1/(t_S+t_{CO})$			50.0			31.2			30.3			25.0	MHz
F_{MAX}	Internal Feedback $1/(t_{SF}+t_{CF})$			80.0			50.0			40.0			30.0	MHz
	No Feedback $1/(t_P)$			83.3			50.0			41.6			33.3	MHz
t_{AW}	Asynchronous Reset Width	15	8		20	9		25	10		30	15		ns
t_{AR}	Asynchronous Reset Recovery Time	15	8		20	12		25	15		30	18		ns
t_{AP}	Asynchronous Reset to Registered Output Reset		12	20		15	22		18	25		20	30	ns

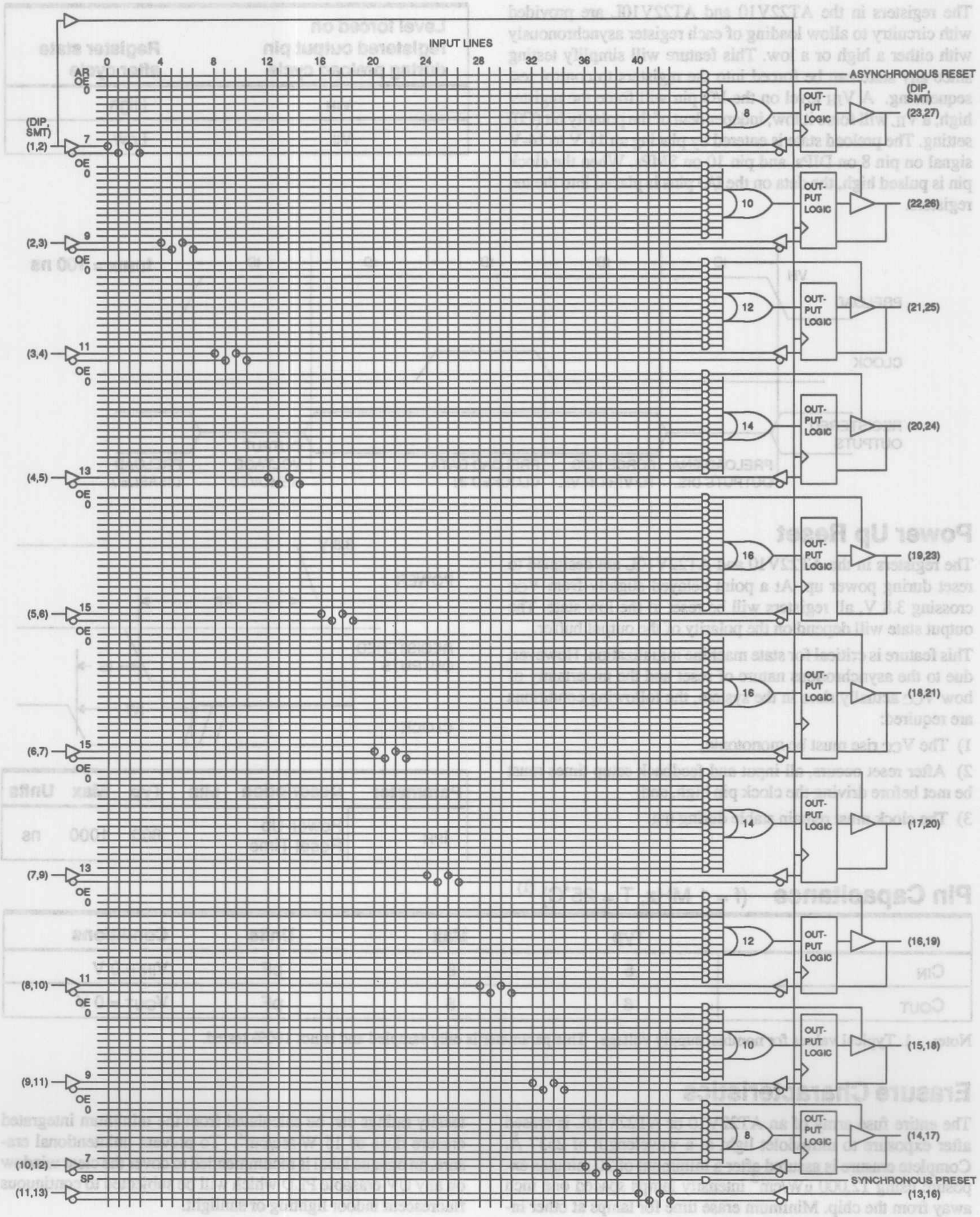
Input Test Waveforms and Measurement Levels



t_R, t_F < 5 ns (10% to 90%)

Output Test Loads:

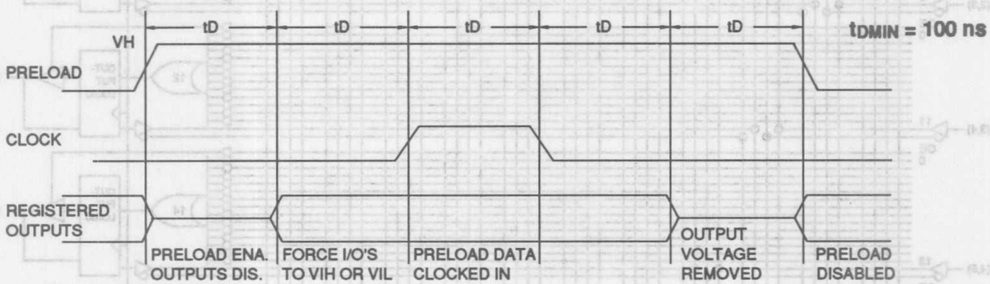
Functional Logic Diagram AT22V10/L



Preload of Registered Outputs

The registers in the AT22V10 and AT22V10L are provided with circuitry to allow loading of each register asynchronously with either a high or a low. This feature will simplify testing since any state can be forced into the registers to control test sequencing. A V_{IH} level on the I/O pin will force the register high; a V_{IL} will force it low, independent of the polarity bit (C0) setting. The preload state is entered by placing an 11-V to 14-V signal on pin 8 on DIPs, and pin 10 on SMPs. When the clock pin is pulsed high, the data on the I/O pins is placed into the ten registers.

Level forced on registered output pin during preload cycle	Register state after cycle
V_{IH}	High
V_{IL}	Low



Power Up Reset

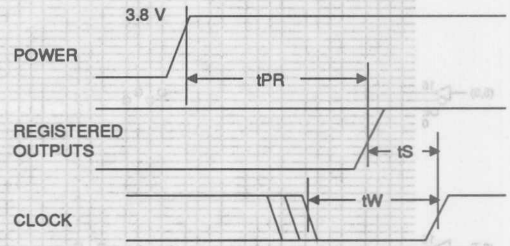
The registers in the AT22V10 and AT22V10L are designed to reset during power up. At a point delayed slightly from V_{CC} crossing 3.8 V, all registers will be reset to the low state. The output state will depend on the polarity of the output buffer.

This feature is critical for state machine initialization. However, due to the asynchronous nature of reset and the uncertainty of how V_{CC} actually rises in the system, the following conditions are required:

1) The V_{CC} rise must be monotonic,

2) After reset occurs, all input and feedback setup times must be met before driving the clock pin high, and

3) The clock must remain stable during t_{PR}.



Parameter	Description	Min	Typ	Max	Units
t_{PR}	Power-Up Reset Time		600	1000	ns

Pin Capacitance (f = 1 MHz, T = 25°C) [1]

	Typ	Max	Units	Conditions
C_{IN}	5	8	pF	V_{IN} = 0 V
C_{OUT}	6	8	pF	V_{OUT} = 0 V

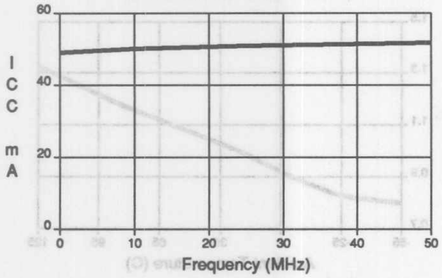
Note: 1. Typical values for nominal supply voltage. This parameter is only sampled and is not 100% tested.

Erasure Characteristics

The entire fuse array of an AT22V10 or AT22V10L is erased after exposure to ultraviolet light at a wavelength of 2537 Å. Complete erasure is assured after a minimum of 20 minutes exposure using 12,000 µW/cm^2 intensity lamps spaced one inch away from the chip. Minimum erase time for lamps at other intensity ratings can be calculated from the minimum integrated erasure dose of 15 W·sec/cm^2. To prevent unintentional erasure, an opaque label is recommended to cover the clear window on any UV erasable PLD which will be subjected to continuous fluorescent indoor lighting or sunlight.

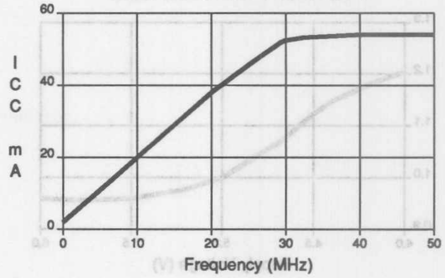
SUPPLY CURRENT vs. INPUT FREQUENCY

AT22V10 (TA = 25C, VCC = 5V)

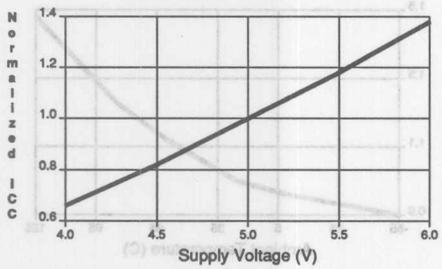


SUPPLY CURRENT vs. INPUT FREQUENCY

AT22V10L (TA = 25C, VCC = 5V)

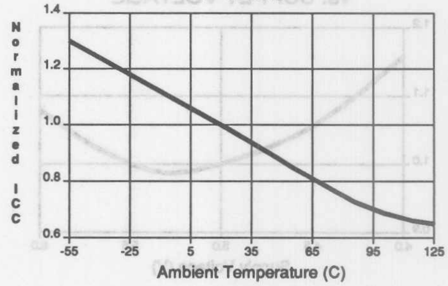


NORMALIZED SUPPLY CURRENT vs. SUPPLY VOLTAGE



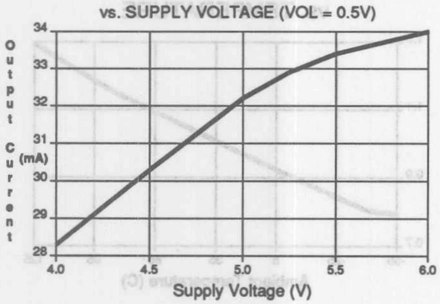
NORMALIZED ICC vs. AMBIENT TEMP.

f = 30 MHz



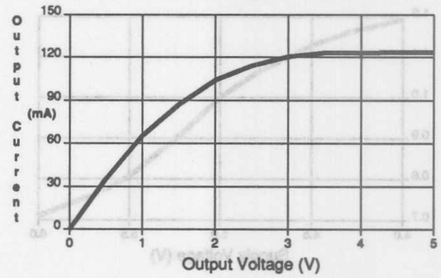
OUTPUT SINK CURRENT

vs. SUPPLY VOLTAGE (VOL = 0.5V)



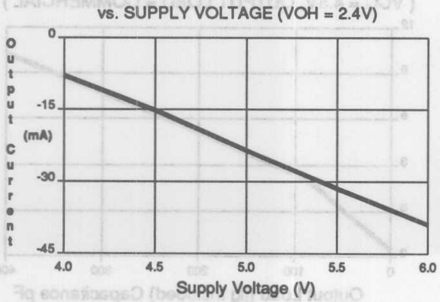
OUTPUT SINK CURRENT

vs. OUTPUT VOLTAGE (TA = 25C, VCC = 5V)



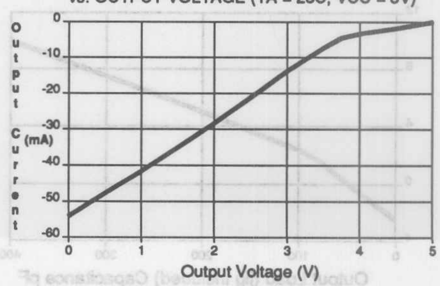
OUTPUT SOURCE CURRENT

vs. SUPPLY VOLTAGE (VOH = 2.4V)

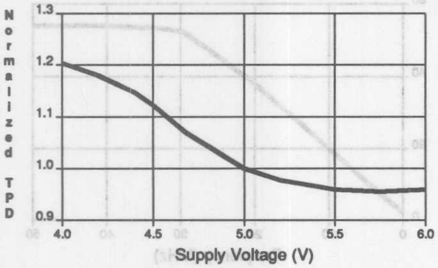


OUTPUT SOURCE CURRENT

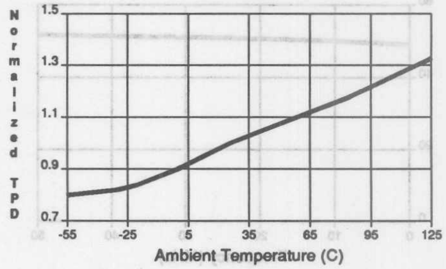
vs. OUTPUT VOLTAGE (TA = 25C, VCC = 5V)



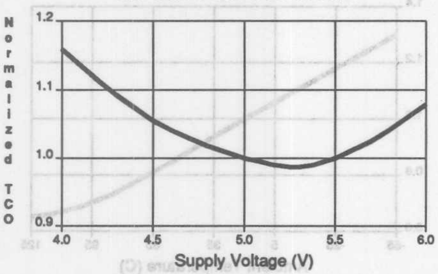
NORMALIZED TPD vs. SUPPLY VOLTAGE



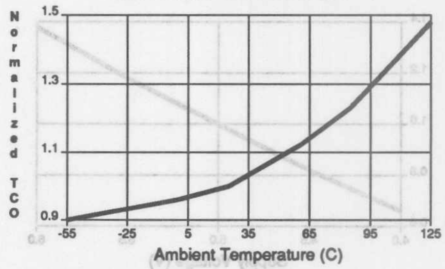
NORMALIZED TPD vs. TEMPERATURE



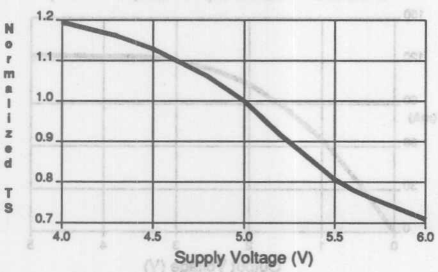
NORMALIZED TCO vs. SUPPLY VOLTAGE



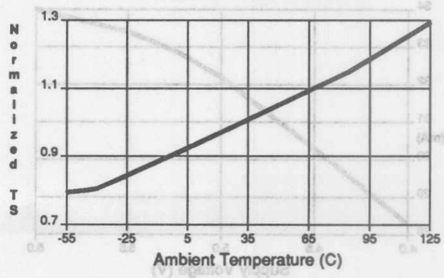
NORMALIZED TCO vs. TEMPERATURE



NORMALIZED TS vs. SUPPLY VOLTAGE

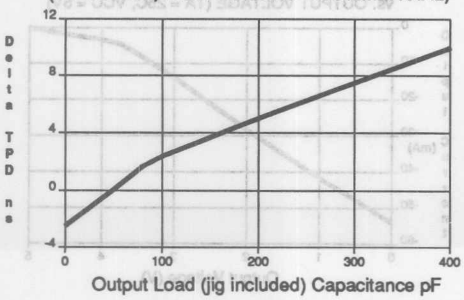


NORMALIZED TS vs. TEMPERATURE



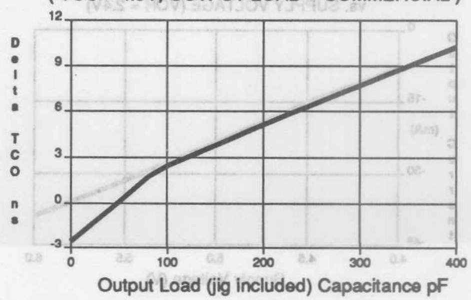
DELTA TPD vs. OUTPUT LOADING

(VCC = 4.5V, OUTPUT LOAD = COMMERCIAL)



DELTA TCO vs. OUTPUT LOADING

(VCC = 4.5V, OUTPUT LOAD = COMMERCIAL)



Ordering Information

t_{PD} (ns)	t_S (ns)	t_{CO} (ns)	Ordering Code	Package	Operation Range
15	10	10	AT22V10-15DC	24DW3	Commercial (0°C to 70°C)
			AT22V10-15GC	24D3	
			AT22V10-15JC	28J	
			AT22V10-15KC	28KW	
			AT22V10-15LC	28LW	
			AT22V10-15NC	28L	
			AT22V10-15PC	24P3	
			AT22V10-15SC	24S	
			AT22V10-15DI	24DW3	Industrial (-40°C to 85°C)
			AT22V10-15GI	24D3	
			AT22V10-15JI	28J	
			AT22V10-15KI	28KW	
			AT22V10-15LI	28LW	
			AT22V10-15NI	28L	
			AT22V10-15PI	24P3	
			AT22V10-15SI	24S	
			AT22V10-15DM	24DW3	Military (-55°C to 125°C)
			AT22V10-15GM	24D3	
			AT22V10-15KM	28KW	
			AT22V10-15LM	28LW	
			AT22V10-15NM	28L	
			AT22V10-15DM/883	24DW3	Military/883C (-55°C to 125°C) Class B, Fully Compliant
			AT22V10-15GM/883	24D3	
			AT22V10-15KM/883	28KW	
			AT22V10-15LM/883	28LW	
			AT22V10-15NM/883	28L	
20	12	15	AT22V10-20DC	24DW3	Commercial (0°C to 70°C)
			AT22V10-20GC	24D3	
			AT22V10-20JC	28J	
			AT22V10-20KC	28KW	
			AT22V10-20LC	28LW	
			AT22V10-20NC	28L	
			AT22V10-20PC	24P3	
			AT22V10-20SC	24S	
			AT22V10-20DI	24DW3	Industrial (-40°C to 85°C)
			AT22V10-20GI	24D3	
			AT22V10-20JI	28J	
			AT22V10-20KI	28KW	
			AT22V10-20LI	28LW	
			AT22V10-20NI	28L	
			AT22V10-20PI	24P3	
			AT22V10-20SI	24S	
20	17	15	AT22V10-20DM	24DW3	Military (-55°C to 125°C)
			AT22V10-20GM	24D3	
			AT22V10-20KM	28KW	
			AT22V10-20LM	28LW	
			AT22V10-20NM	28L	

Ordering Information

t_{PD} (ns)	t_S (ns)	t_{CO} (ns)	Ordering Code	Package	Operation Range
20	17	15	AT22V10-20DM/883 AT22V10-20GM/883 AT22V10-20KM/883 AT22V10-20LM/883 AT22V10-20NM/883	24DW3 24D3 28KW 28LW 28L	Military/883C (-55°C to 125°C) Class B, Fully Compliant
25	15	15	AT22V10-25DC AT22V10-25GC AT22V10-25JC AT22V10-25KC AT22V10-25LC AT22V10-25NC AT22V10-25PC AT22V10-25SC	24DW3 24D3 28J 28KW 28LW 28L 24P3 24S	Commercial (0°C to 70°C)
			AT22V10-25DI AT22V10-25GI AT22V10-25JI AT22V10-25KI AT22V10-25LI AT22V10-25NI AT22V10-25PI AT22V10-25SI	24DW3 24D3 28J 28KW 28LW 28L 24P3 24S	Industrial (-40°C to 85°C)
25	18	15	AT22V10-25DM AT22V10-25GM AT22V10-25KM AT22V10-25LM AT22V10-25NM	24DW3 24D3 28KW 28LW 28L	Military (-55°C to 125°C)
			AT22V10-25DM/883 AT22V10-25GM/883 AT22V10-25KM/883 AT22V10-25LM/883 AT22V10-25NM/883	24DW3 24D3 28KW 28LW 28L	Military/883C (-55°C to 125°C) Class B, Fully Compliant
30	20	20	AT22V10-30DM AT22V10-30GM AT22V10-30KM AT22V10-30LM AT22V10-30NM	24DW3 24D3 28KW 28LW 28L	Military (-55°C to 125°C)
			AT22V10-30DM/883 AT22V10-30GM/883 AT22V10-30KM/883 AT22V10-30LM/883 AT22V10-30NM/883	24DW3 24D3 28KW 28LW 28L	Military/883C (-55°C to 125°C) Class B, Fully Compliant
20	17	15	5962-87539 04 LX 5962-87539 04 3X	24DW3 28LW	Military/883C (-55°C to 125°C) Class B, Fully Compliant
25	18	15	5962-87539 01 LX 5962-87539 01 3X	24DW3 28LW	Military/883C (-55°C to 125°C) Class B, Fully Compliant

Ordering Information

tPD (ns)	ts (ns)	tCO (ns)	Ordering Code	Package	Operation Range
30	20	20	5962-87539 02 LX 5962-87539 02 3X	24DW3 28LW	Military/883C (-55°C to 125°C) Class B, Fully Compliant
15	10	10	5962-88670 05 LX 5962-88670 05 3X	24D3 28L	Military/883C (-55°C to 125°C) Class B, Fully Compliant
20	17	15	5962-88670 04 LX 5962-88670 04 3X	24D3 28L	Military/883C (-55°C to 125°C) Class B, Fully Compliant
25	18	15	5962-88670 01 LX 5962-88670 01 3X	24D3 28L	Military/883C (-55°C to 125°C) Class B, Fully Compliant
30	20	20	5962-88670 02 LX 5962-88670 02 3X	24D3 28L	Military/883C (-55°C to 125°C) Class B, Fully Compliant

Ordering Information

t_{PD} (ns)	t_S (ns)	t_{CO} (ns)	Ordering Code	Package	Operation Range
20	12	15	AT22V10L-20DC	24DW3	Commercial (0°C to 70°C)
			AT22V10L-20GC	24D3	
			AT22V10L-20JC	28J	
			AT22V10L-20KC	28KW	
			AT22V10L-20LC	28LW	
			AT22V10L-20NC	28L	
			AT22V10L-20PC	24P3	
			AT22V10L-20SC	24S	
			AT22V10L-20DI	24DW3	Industrial (-40°C to 85°C)
			AT22V10L-20GI	24D3	
			AT22V10L-20JI	28J	
			AT22V10L-20KI	28KW	
			AT22V10L-20LI	28LW	
			AT22V10L-20NI	28L	
			AT22V10L-20PI	24P3	
			AT22V10L-20SI	24S	
20	17	15	AT22V10L-20DM	24DW3	Military (-55°C to 125°C)
			AT22V10L-20GM	24D3	
			AT22V10L-20KM	28KW	
			AT22V10L-20LM	28LW	
			AT22V10L-20NM	28L	
			AT22V10L-20DM/883	24DW3	Military/883C (-55°C to 125°C) Class B, Fully Compliant
			AT22V10L-20GM/883	24D3	
			AT22V10L-20KM/883	28KW	
			AT22V10L-20LM/883	28LW	
			AT22V10L-20NM/883	28L	
25	15	15	AT22V10L-25DC	24DW3	Commercial (0°C to 70°C)
			AT22V10L-25GC	24D3	
			AT22V10L-25JC	28J	
			AT22V10L-25KC	28KW	
			AT22V10L-25LC	28LW	
			AT22V10L-25NC	28L	
			AT22V10L-25PC	24P3	
			AT22V10L-25SC	24S	
			AT22V10L-25DI	24DW3	Industrial (-40°C to 85°C)
			AT22V10L-25GI	24D3	
			AT22V10L-25JI	28J	
			AT22V10L-25KI	28KW	
			AT22V10L-25LI	28LW	
			AT22V10L-25NI	28L	
			AT22V10L-25PI	24P3	
			AT22V10L-25SI	24S	
25	18	15	AT22V10L-25DM	24DW3	Military (-55°C to 125°C)
			AT22V10L-25GM	24D3	
			AT22V10L-25KM	28KW	
			AT22V10L-25LM	28LW	
			AT22V10L-25NM	28L	

Ordering Information

tPD (ns)	ts (ns)	tCO (ns)	Ordering Code	Package	Operation Range
25	18	15	AT22V10L-25DM/883 AT22V10L-25GM/883 AT22V10L-25KM/883 AT22V10L-25LM/883 AT22V10L-25NM/883	24DW3 24D3 28KW 28LW 28L	Military/883C (-55°C to 125°C) Class B, Fully Compliant
30	20	20	AT22V10L-30DM AT22V10L-30GM AT22V10L-30KM AT22V10L-30LM AT22V10L-30NM	24DW3 24D3 28KW 28LW 28L	Military (-55°C to 125°C)
			AT22V10L-30DM/883 AT22V10L-30GM/883 AT22V10L-30KM/883 AT22V10L-30LM/883 AT22V10L-30NM/883	24DW3 24D3 28KW 28LW 28L	Military/883C (-55°C to 125°C) Class B, Fully Compliant
20	17	15	5962-88724 04 LX 5962-88724 04 3X	24DW3 28L	Military/883C (-55°C to 125°C) Class B, Fully Compliant
25	18	15	5962-88724 01 LX 5962-88724 01 3X	24DW3 28LW	Military/883C (-55°C to 125°C) Class B, Fully Compliant
30	20	20	5962-88724 02 LX 5962-88724 02 3X	24DW3 28LW	Military/883C (-55°C to 125°C) Class B, Fully Compliant
20	17	15	5962-89755 04 LX 5962-89755 04 3X	24D3 28L	Military/883C (-55°C to 125°C) Class B, Fully Compliant
25	18	15	5962-89755 01 LX 5962-89755 01 3X	24D3 28L	Military/883C (-55°C to 125°C) Class B, Fully Compliant
30	20	20	5962-89755 02 LX 5962-89755 02 3X	24D3 28L	Military/883C (-55°C to 125°C) Class B, Fully Compliant

Package Type	
24DW3	24 Lead, 0.300" Wide, Windowed, Ceramic Dual Inline Package (Cerdip)
24D3	24 Lead, 0.300" Wide, Non-Windowed (OTP), Ceramic Dual Inline Package (Cerdip)
28J	28 Lead, Plastic J-Leaded Chip Carrier OTP (PLCC)
28KW	28 Lead, Windowed, Ceramic J-Leaded Chip Carrier (JLCC)
28LW	28 Pad, Windowed, Ceramic Leadless Chip Carrier (LCC)
28L	28 Pad, Non-Windowed, Ceramic Leadless Chip Carrier OTP (LCC)
24P3	24 Lead, 0.300" Wide, Plastic Dual Inline Package OTP (PDIP)
24S	24 Lead, 0.300" Wide, Plastic Gull Wing Small Outline OTP (SOIC)

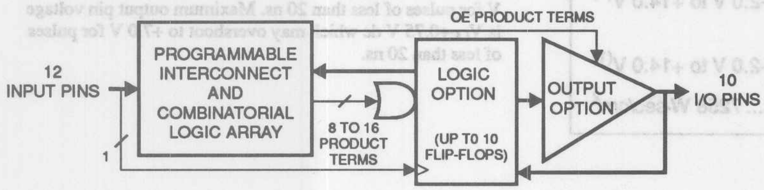
1

High Speed UV Erasable Programmable Logic Device

Features

- **High Performance Programmable Logic Device**
  - **7.5 ns Max Propagation Delay**
  - **Up to 166 MHz Operation**
  - **5 V ± 10% Operation**
- **Fully Compatible with Standard 22V10**
  - **Identical Functionality/Fuse-Map**
- **TTL Compatible Inputs and Outputs**
  - **10 μA Leakage Maximum**
- **Reprogrammable - Tested 100% for Programmability**
- **High Reliability**
  - **Proven UV Erasable CMOS Technology**
  - **2000 V ESD Protection**
  - **200 mA Latch-Up Protection**
- **Full Military, Commercial and Industrial Temperature Ranges**
- **Dual-In-Line and Surface Mount Packages with Standard Pinouts**

Logic Diagram



Description

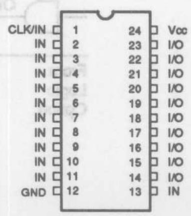
The AT22V10B is an ultra-high performance CMOS Programmable Logic Device (PLD). Speeds down to 7.5 ns and operation up to 166 MHz are offered. All pins offer a low ± 10 μA leakage.

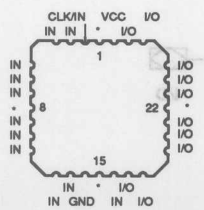
The AT22V10B logic functionality is fully compatible with the standard 22V10. The 12 dedicated inputs and ten configurable I/O pins allow implementation of logic requiring up to 22 input signals. The AT22V10B also provides individual output enable product terms for each of the ten I/Os.

Continued on next page.

Pin Configurations

Pin Name	Function
CLK/IN	Clock and Logic Input
IN	Logic Inputs
I/O	Bidirectional Buffers
*	No Internal Connection
VCC	+5 V Supply





Description (Continued)

The AT22V10B incorporates a variable product term architecture. Each output is allocated from eight to 16 product terms, which allows highly complex logic functions to be realized.

The AT22V10B includes two additional product terms to provide synchronous preset and asynchronous reset. These terms are common to all ten registers. All registers are automatically cleared upon power up.

Register preload simplifies testing. A security fuse prevents unauthorized copying of programmed fuse patterns.

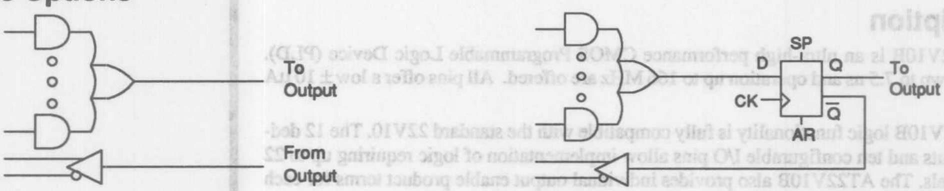
Absolute Maximum Ratings*

Temperature Under Bias	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-2.0 V to +7.0 V(1)
Voltage on Input Pins with Respect to Ground During Programming	-2.0 V to +14.0 V(1)
Programming Voltage with Respect to Ground	-2.0 V to +14.0 V(1)
Integrated UV Erase Dose	7258 W-sec/cm^2

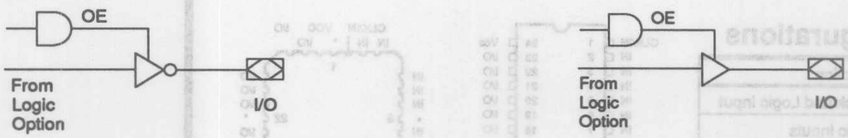
*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Note:
1. Minimum voltage is -0.6 V dc which may undershoot to -2.0 V for pulses of less than 20 ns. Maximum output pin voltage is V_{CC}+0.75 V dc which may overshoot to +7.0 V for pulses of less than 20 ns.

Logic Options



Output Options



D.C. and A.C. Operating Conditions

	Commercial AT22V10B -7	Commercial AT22V10B -10, -12	Industrial AT22V10B -10, -12	Military AT22V10B -10, -12
Operating Temperature (Case)	0°C - 70°C	0°C - 70°C	-40°C - 85°C	-55°C - 125°C
V_{CC} Power Supply	5 V ± 5%	5 V ± 10%	5 V ± 10%	5 V ± 10%

Operating Modes

Mode	24-Pin DIP / 28- Pin JLCC	1 / 2	5 / 6	8 / 10	13 / 16	I/O's / I/O's	V_{CC}(24) / V_{CC}(28)
"PLD"		X [1]	X	X	X	I/O	5V
Program		V_{PP}	X / V_H [2]	X	V_{PP}	D_{IN}	6V
PGM Verify		V_{PP}	X/V_H	X	V_{IL}	D_{OUT}	6V
PGM Inhibit		V_{PP}	X/V_H	X	V_{IH}	High Z	6V
Preload		X	X	V_H	X	D_{IN}	5V

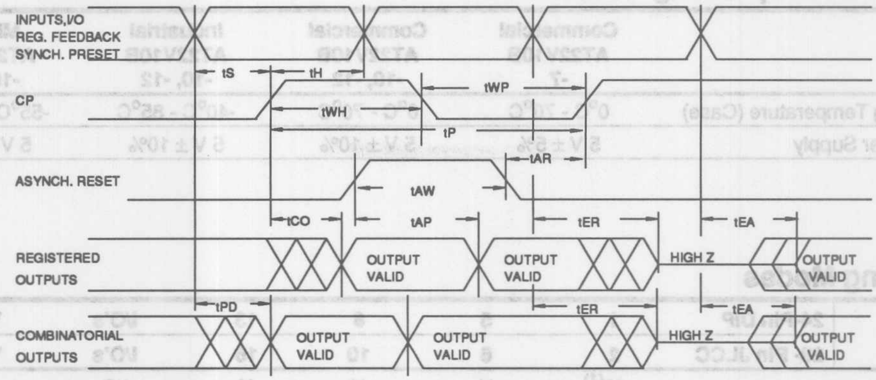
Notes: 1. X can be V_{IL} or V_{IH}. 2. V_H = 11.0 V to 14.0 V

D.C. Characteristics

Symbol	Parameter	Condition			Min	Max	Units
I_{LI}	Input Load Current	V_{IN} = -0.1 V to V_{CC}+1 V				10	µA
I_{LO}	Output Leakage Current	V_{OUT} = -0.1 V to V_{CC}+0.1 V				10	µA
I_{CC}	Power Supply Current	f = 0 MHz to F_{MAX}, V_{CC} = MAX, V_{IN} = GND, Outputs Open		Com.		140	mA
				Ind., Mil.		160	mA
I_{OS} [1]	Output Short Circuit Current	V_{OUT} = 0.5 V			-30	-120	mA
V_{IL}	Input Low Voltage				-0.6	0.8	V
V_{IH}	Input High Voltage				2.0	V_{CC}+0.75	V
V_{OL}	Output Low Voltage	V_{IN} = V_{IH} or V_{IL}, V_{CC} = MIN	I_{OL} = 16 mA	Com.,Ind.		0.5	V
			I_{OL} = 12 mA	Mil.		0.5	V
			I_{OL} = 24 mA	Com.		0.8	V
V_{OH}	Output High Voltage	V_{IN} = V_{IH} or V_{IL}, V_{CC} = MIN	I_{OH} = -4.0 mA		2.4		V

Notes: 1. Not more than one output at a time should be shorted. Duration of short circuit test should not exceed 30 sec.

A.C. Waveforms [1]



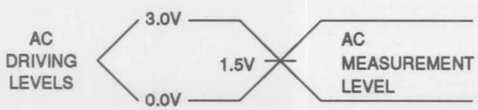
Note: 1. Timing measurement reference is 1.5 V. Input AC driving levels are 0.0 V and 3.0 V, unless otherwise specified.

A.C. Characteristics

Symbol	Parameter	AT22V10B-7 Min	AT22V10B-7 Typ	AT22V10B-7 Max	AT22V10B-10 Min	AT22V10B-10 Typ	AT22V10B-10 Max	AT22V10B-12 Min	AT22V10B-12 Typ	AT22V10B-12 Max	Units
t_{PD}	Input or Feedback to Non-Registered Output		5	7.5		6	10		7	12	ns
t_{EA}	Input to Output Enable		5	7.5		6	10		7	12	ns
t_{ER}	Input to Output Disable		5	7.5		6	10		7	12	ns
t_{CF} (1)	Clock to Feedback	0	1	2	0	1	2	0	1.5	2	ns
t_{CO}	Clock to Output	0	3.5	5.5	0	4	7	0	5	7.5	ns
t_S	Input or Feedback Setup Time	3.5	2		5	3		6	5		ns
t_H	Hold Time	0			0			0			ns
t_P	Clock Period	6			7			8			ns
t_{WL} (1)	Clock Width Low	3			3.5			4			ns
t_{WH}	Clock Width High	3			3.5			4			ns
F_{MAX}	External Feedback $1/(t_S+t_{CO})$			111			83			74	MHz
	Internal Feedback $1/(t_S + t_{CF})$			166			142			125	MHz
	No Feedback $1/(t_P)$			166			142			125	MHz
t_{AW}	Asynchronous Reset Width	6	3		7	4		8	5		ns
t_{AR}	Asynchronous Reset, Synchronous Preset, Recovery Time	7	4		8	5		8.5	5		ns
t_{AP}	Asynchronous Reset to Registered Output Reset		6	10		8	14		9	15	ns

Note: 1. This parameter is only sampled and is not 100% tested.

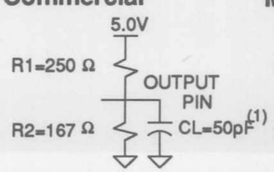
Input Test Waveforms and Measurement Levels



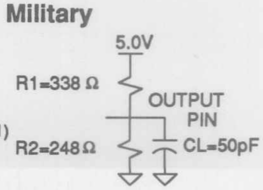
t_R, t_F < 2 ns (10% to 90%)

Output Test Loads:

Commercial

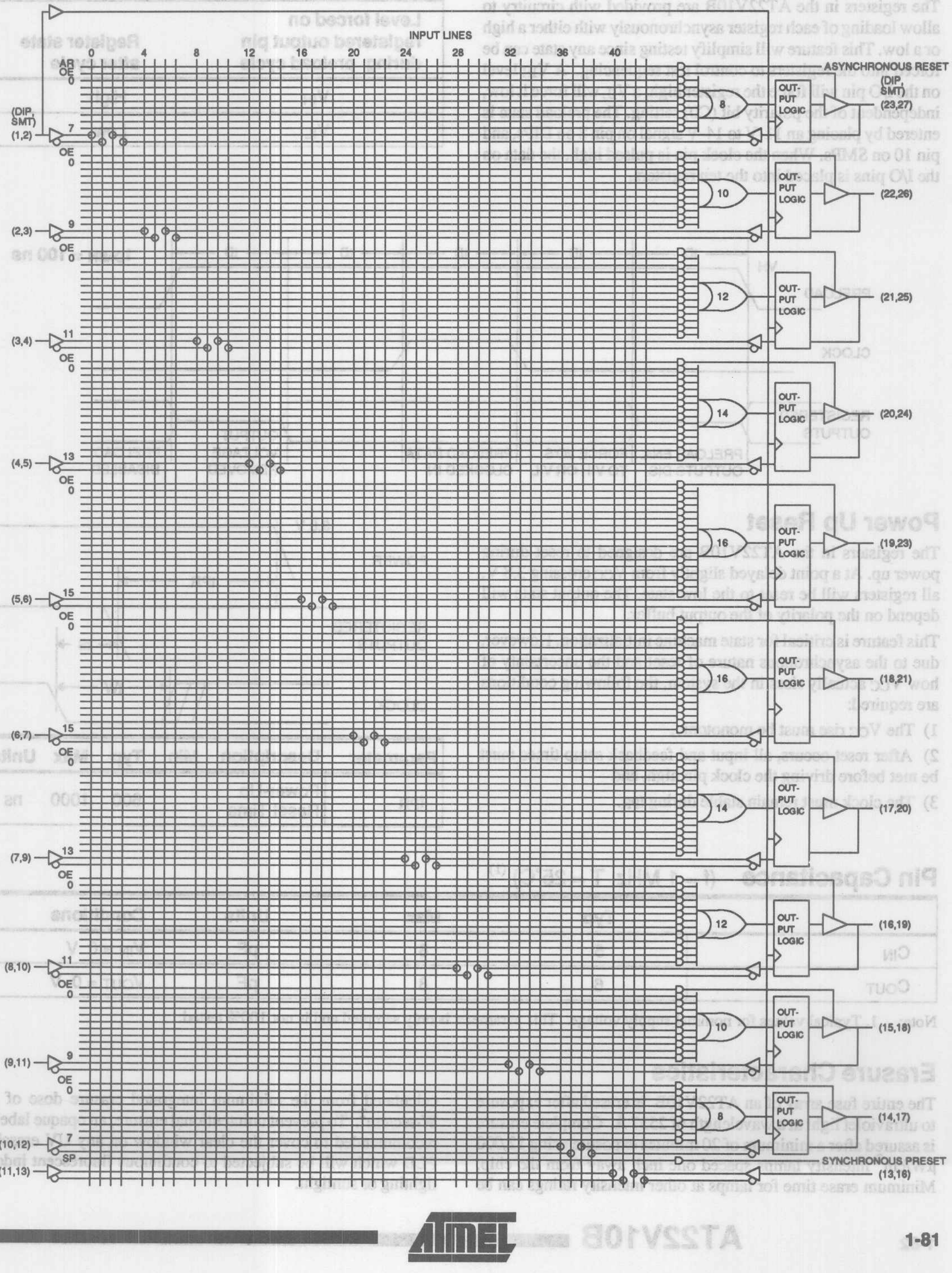


Military



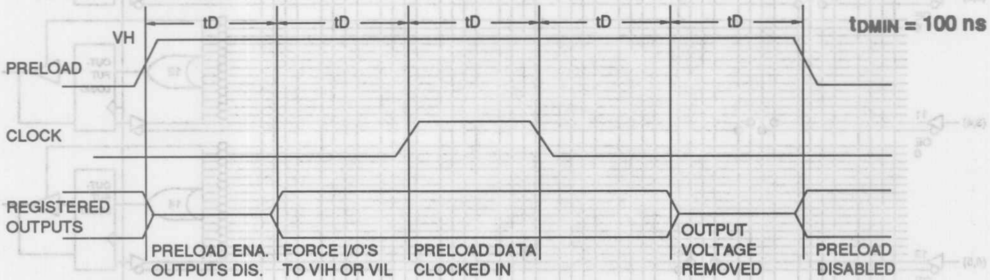
Note: 1. CL = 30 pF for AT22V10B-7

Functional Logic Diagram AT22V10B



The registers in the AT22V10B are provided with circuitry to allow loading of each register asynchronously with either a high or a low. This feature will simplify testing since any state can be forced into the registers to control test sequencing. A V_{IH} level on the I/O pin will force the register high; a V_{IL} will force it low, independent of the polarity bit (C0) setting. The preload state is entered by placing an 11-V to 14-V signal on pin 8 on DIPs, and pin 10 on SMPs. When the clock pin is pulsed high, the data on the I/O pins is placed into the ten registers.

Level forced on registered output pin during preload cycle	Register state after cycle
V_{IH}	High
V_{IL}	Low



Power Up Reset

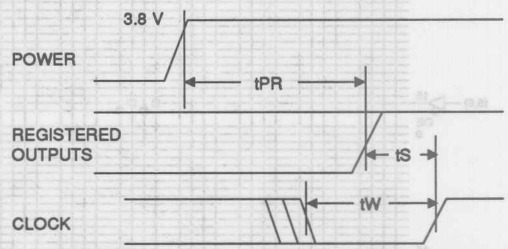
The registers in the AT22V10B are designed to reset during power up. At a point delayed slightly from V_{CC} crossing 3.8 V, all registers will be reset to the low state. The output state will depend on the polarity of the output buffer.

This feature is critical for state machine initialization. However, due to the asynchronous nature of reset and the uncertainty of how V_{CC} actually rises in the system, the following conditions are required:

1) The V_{CC} rise must be monotonic,

2) After reset occurs, all input and feedback setup times must be met before driving the clock pin high, and

3) The clock must remain stable during t_{PR}.



Parameter	Description	Min	Typ	Max	Units
t_{PR}	Power-Up Reset Time		600	1000	ns

Pin Capacitance (f = 1 MHz, T = 25°C) [1]

	Typ	Max	Units	Conditions
C_{IN}	5	8	pF	V_{IN} = 0 V
C_{OUT}	6	8	pF	V_{OUT} = 0 V

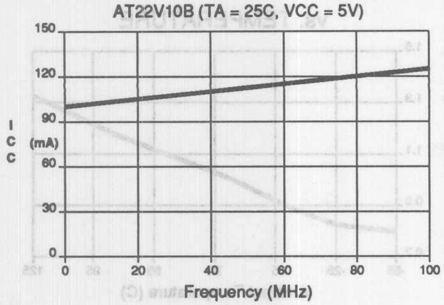
Note: 1. Typical values for nominal supply voltage. This parameter is only sampled and is not 100% tested.

Erasure Characteristics

The entire fuse array of an AT22V10B is erased after exposure to ultraviolet light at a wavelength of 2537 Å. Complete erasure is assured after a minimum of 20 minutes exposure using 12,000 $\mu W/cm^2$ intensity lamps spaced one inch away from the chip. Minimum erase time for lamps at other intensity ratings can be calculated from the minimum integrated erasure dose of 15 $W \cdot sec/cm^2$. To prevent unintentional erasure, an opaque label is recommended to cover the clear window on any UV erasable PLD which will be subjected to continuous fluorescent indoor lighting or sunlight.

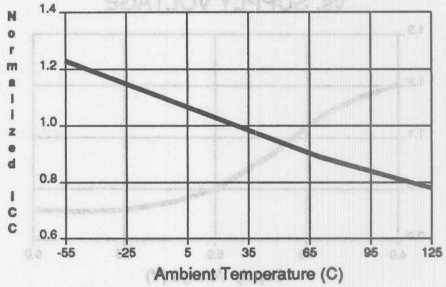
SUPPLY CURRENT vs. INPUT FREQUENCY

AT22V10B (TA = 25C, VCC = 5V)

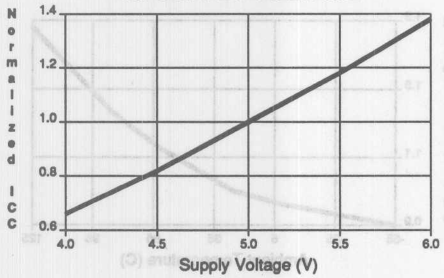


NORMALIZED ICC vs. AMBIENT TEMP.

f = 50 MHz

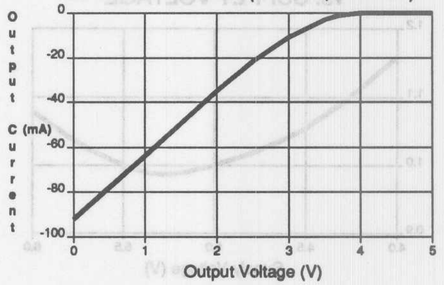


NORMALIZED SUPPLY CURRENT vs. SUPPLY VOLTAGE



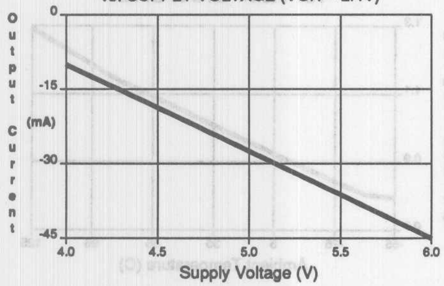
OUTPUT SOURCE CURRENT

vs. OUTPUT VOLTAGE (TA = 25C, VCC = 5V)



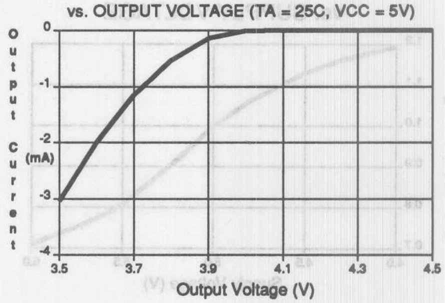
OUTPUT SOURCE CURRENT

vs. SUPPLY VOLTAGE (VOH = 2.4V)



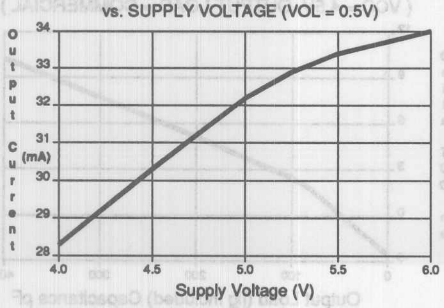
OUTPUT SOURCE CURRENT

vs. OUTPUT VOLTAGE (TA = 25C, VCC = 5V)



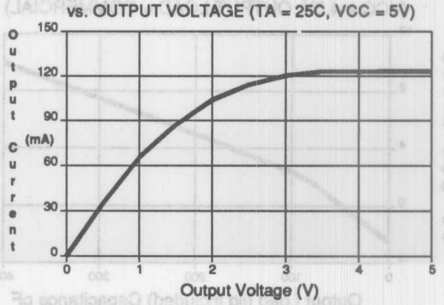
OUTPUT SINK CURRENT

vs. SUPPLY VOLTAGE (VOL = 0.5V)

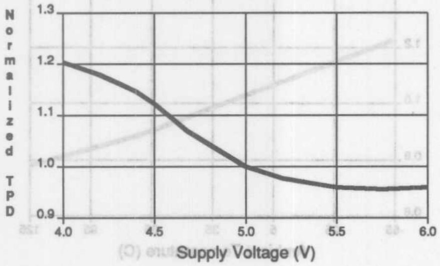


OUTPUT SINK CURRENT

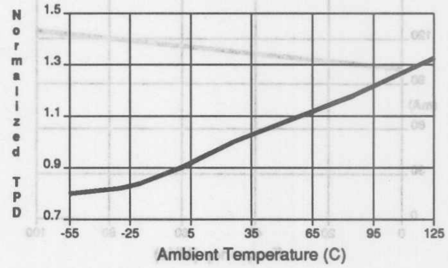
vs. OUTPUT VOLTAGE (TA = 25C, VCC = 5V)



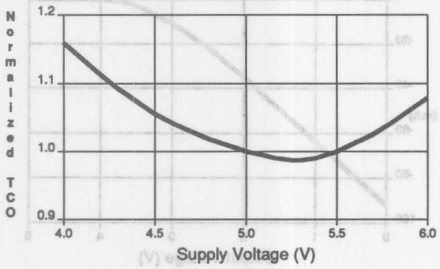
NORMALIZED TPD vs. SUPPLY VOLTAGE



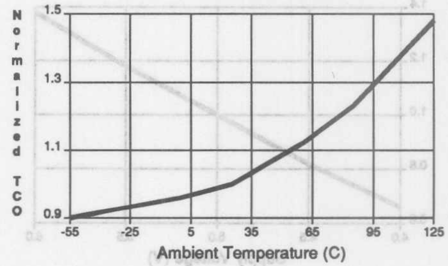
NORMALIZED TPD vs. TEMPERATURE



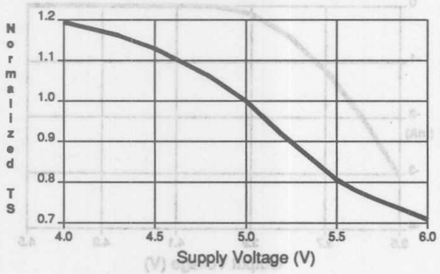
NORMALIZED TCO vs. SUPPLY VOLTAGE



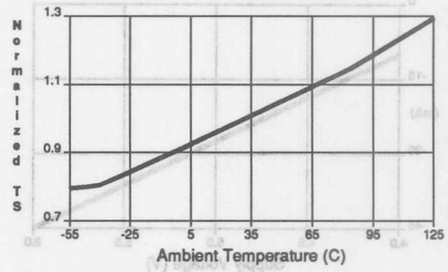
NORMALIZED TCO vs. TEMPERATURE



NORMALIZED TS vs. SUPPLY VOLTAGE

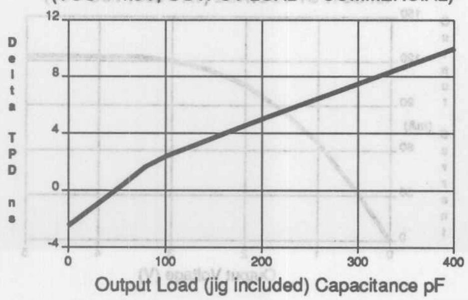


NORMALIZED TS vs. TEMPERATURE



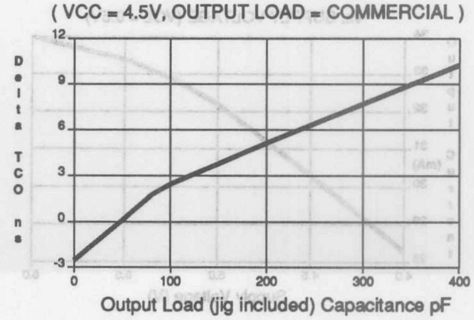
DELTA TPD vs. OUTPUT LOADING

(VCC = 4.5V, OUTPUT LOAD = COMMERCIAL)



DELTA TCO vs. OUTPUT LOADING

(VCC = 4.5V, OUTPUT LOAD = COMMERCIAL)



tPD (ns)	ts (ns)	tCO (ns)	Ordering Code	Package	Operation Range
7.5	3.5	5.5	AT22V10B-7DC	24DW3	Commercial (0°C to 70°C)
			AT22V10B-7JC	28J	
			AT22V10B-7KC	28KW	
			AT22V10B-7PC	24P3	
10	5	7	AT22V10B-10DC	24DW3	Commercial (0°C to 70°C)
			AT22V10B-10GC	24D3	
			AT22V10B-10JC	28J	
			AT22V10B-10KC	28KW	
			AT22V10B-10LC	28LW	
			AT22V10B-10NC	28L	
			AT22V10B-10PC	24P3	
			AT22V10B-10SC	24S	
			AT22V10B-10DI	24DW3	Industrial (-40°C to 85°C)
			AT22V10B-10GI	24D3	
			AT22V10B-10JI	28J	
			AT22V10B-10KI	28KW	
			AT22V10B-10LI	28LW	
			AT22V10B-10NI	28L	
			AT22V10B-10PI	24P3	
			AT22V10B-10SI	24S	
			AT22V10B-10DM	24DW3	Military (-55°C to 125°C)
			AT22V10B-10GM	24D3	
			AT22V10B-10KM	28KW	
			AT22V10B-10LM	28LW	
			AT22V10B-10NM	28L	
			AT22V10B-10DM/883	24DW3	Military/883C (-55°C to 125°C) Class B, Fully Compliant
			AT22V10B-10GM/883	24D3	
			AT22V10B-10KM/883	28KW	
			AT22V10B-10LM/883	28LW	
			AT22V10B-10NM/883	28L	
12	6	7.5	AT22V10B-12DC	24DW3	Commercial (0°C to 70°C)
			AT22V10B-12GC	24D3	
			AT22V10B-12JC	28J	
			AT22V10B-12KC	28KW	
			AT22V10B-12LC	28LW	
			AT22V10B-12NC	28L	
			AT22V10B-12PC	24P3	
			AT22V10B-12SC	24S	
			AT22V10B-12DI	24DW3	Industrial (-40°C to 85°C)
			AT22V10B-12GI	24D3	
			AT22V10B-12JI	28J	
			AT22V10B-12KI	28KW	
			AT22V10B-12LI	28LW	
			AT22V10B-12NI	28L	
			AT22V10B-12PI	24P3	
			AT22V10B-12SI	24S	

Ordering Information

t_{PD} (ns)	t_S (ns)	t_{CO} (ns)	Ordering Code	Package	Operation Range
12	6	7.5	AT22V10B-12DM AT22V10B-12GM AT22V10B-12KM AT22V10B-12LM AT22V10B-12NM	24DW3 24D3 28KW 28LW 28L	Military (-55°C to 125°C)
			AT22V10B-12DM/883 AT22V10B-12GM/883 AT22V10B-12KM/883 AT22V10B-12LM/883 AT22V10B-12NM/883	24DW3 24D3 28KW 28LW 28L	Military/883C (-55°C to 125°C) Class B, Fully Compliant
10	5	7	5962-87539 06 LX 5962-87539 06 3X 5962-87539 06 XX	24DW3 28L 28KW	Military/883C (-55°C to 125°C) Class B, Fully Compliant

Package Type	
24DW3	24 Lead, 0.300" Wide, Windowed, Ceramic Dual Inline Package (Cerdip)
24D3	24 Lead, 0.300" Wide, Non-Windowed (OTP), Ceramic Dual Inline Package (Cerdip)
28J	28 Lead, Plastic J-Leaded Chip Carrier OTP (PLCC)
28KW	28 Lead, Windowed, Ceramic J-Leaded Chip Carrier (JLCC)
28LW	28 Pad, Windowed, Ceramic Leadless Chip Carrier (LCC)
28L	28 Pad, Non-Windowed, Ceramic Leadless Chip Carrier OTP (LCC)
24P3	24 Lead, 0.300" Wide, Plastic Dual Inline Package OTP (PDIP)
24S	24 Lead, 0.300" Wide, Plastic Gull Wing Small Outline OTP (SOIC)

Low Voltage UV Erasable Programmable Logic Device

1

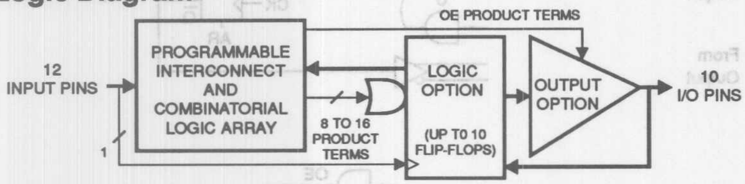
Features

- Low Voltage Programmable Logic Device
  - Wide Power Supply Range - 3.0 V to 5.5 V
  - Ideal for Battery Powered Systems
- High Speed Operation
  - 20 ns max Propagation Delay at V_{CC} = 3.0 V
- Full Military, Commercial and Industrial Temperature Ranges
- Familiar 22V10 Logic Architecture
- Low Power 3-Volt CMOS Operation

	AT22LV10L	AT22LV10	
Temp	Com./Mil.	Com./Mil.	
I_{CC} (mA)	4 / 5	35 / 45	V_{CC} = 3.6 V

- CMOS and TTL Compatible Inputs and Outputs
  - 10 µA Leakage Maximum
- Reprogrammable - Tested 100% for Programmability
- High Reliability CMOS Technology
  - 2000 V ESD Protection
  - 200 mA Latchup Immunity
- Dual-In-Line and Surface Mount Packages

Logic Diagram



Description

The AT22LV10 and AT22LV10L are low voltage compatible CMOS high performance Programmable Logic Devices (PLDs). Speeds down to 20 ns and power dissipation as low as 14.4 mW are offered. All speed ranges are specified over the 3.0 V to 5.5 V range. All pins offer a low ±10 µA leakage.

The AT22LV10L provides the optimum low power CMOS PLD solution, with low DC power (1 mA typical at V_{CC} = 3.3 V) and full CMOS output levels. The AT22LV10L significantly reduces total system power, allowing battery powered operation.

Full CMOS output levels help reduce power in many other system components.

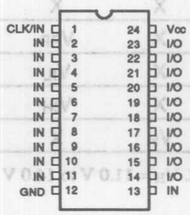
The AT22LV10 and AT22LV10L logic architectures are identical to the familiar 22V10. Each output is allocated from eight to 16 product terms, which allows highly complex logic functions to be realized.

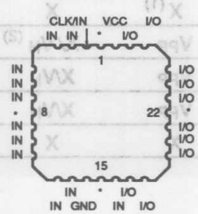
Two additional product terms are included to provide synchronous preset and asynchronous reset. These terms are common to all ten registers. All registers are automatically cleared upon power up.

Register preload simplifies testing. A security fuse prevents unauthorized copying of programmed fuse patterns.

Pin Configurations

Pin Name	Function
CLK/IN	Clock and Logic Input
IN	Logic Inputs
I/O	Bidirectional Buffers
*	No Internal Connection
VCC	3.0 V to 5.5 V Supply





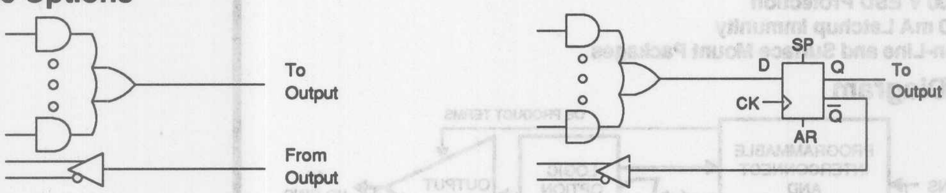
Absolute Maximum Ratings*

Temperature Under Bias	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-2.0 V to +7.0 V(1)
Voltage on Input Pins with Respect to Ground During Programming	-2.0 V to +14.0 V(1)
Programming Voltage with Respect to Ground	-2.0 V to +14.0 V(1)
Integrated UV Erase Dose	7258 W·sec/cm²

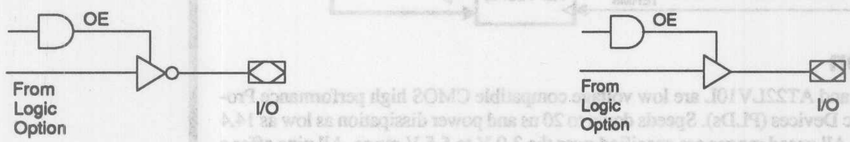
*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Note:
1. Minimum voltage is -0.6 V dc which may undershoot to -2.0 V for pulses of less than 20 ns. Maximum pin voltage is V_{CC}+0.75 V dc which may overshoot to V_{CC} +2.0 V for pulses of less than 20 ns.

Logic Options



Output Options



D.C. and A.C. Operating Conditions

	Commercial AT22LV10/L -20, -25, -30	Industrial AT22LV10/L -20, -25, -30	Military AT22LV10/L -25, -30, -35
Operating Temperature (Case)	0°C - 70°C	-40°C - 85°C	-55°C - 125°C
V_{CC} Power Supply	3.0 V to 5.5 V	3.0 V to 5.5 V	3.0 V to 5.5 V

Operating Modes

Mode	24-Pin DIP	1	5	8	13	I/Os	V_{CC} (24)
	28-Pin JLCC	2	6	10	16	I/Os	V_{CC} (28)
"PLD"		X (1)	X	X	X	I/O	3.0 V to 5.5 V
Program		V_{PP}	X/V_H (2)	X	V_{PP}	D_{IN}	6 V
PGM Verify		V_{PP}	X/V_H	X	V_{IL}	D_{OUT}	6 V
PGM Inhibit		V_{PP}	X/V_H	X	V_{IH}	High Z	6 V
Preload		X	X	V_H	X	D_{IN}	3.0 V to 5.5 V

Notes: 1. X can be V_{IL} or V_{IH}.
2. V_H = 11.0 V to 14.0 V

D.C. Characteristics

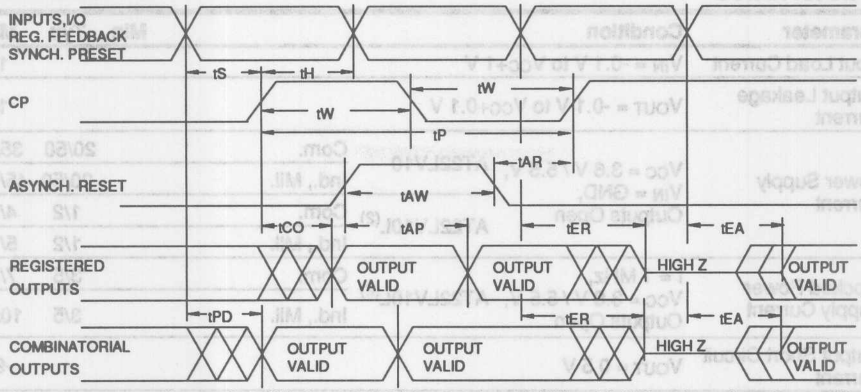
Symbol	Parameter	Condition			Min	Typ	Max	Units
I_{LI}	Input Load Current	V_{IN} = -0.1 V to V_{CC}+1 V					10	µA
I_{LO}	Output Leakage Current	V_{OUT} = -0.1 V to V_{CC}+0.1 V					10	µA
I_{CC}	Power Supply Current	V_{CC} = 3.6 V / 5.5 V, V_{IN} = GND, Outputs Open	AT22LV10	Com.		20/50	35/90	mA
				Ind., Mil.		20/50	45/100	mA
			AT22LV10L[(2)]	Com.		1/2	4/12	mA
				Ind., Mil.		1/2	5/15	mA
I_{CC2}	Clocked Power Supply Current	f = 1 MHz, V_{CC} = 3.6 V / 5.5 V, Outputs Open	AT22LV10L[(2)]	Com.		3/5	7/15	mA
				Ind., Mil.		3/5	10/20	mA
I_{OS} [(1)]	Output Short Circuit Current	V_{OUT} = 0.5 V					-90	mA
V_{IL1}	Input Low Voltage	4.5 V ≤ V_{CC} ≤ 5.5 V			-0.6		0.8	V
V_{IL2}	Input Low Voltage	3.0 V ≤ V_{CC} < 4.5 V			-0.6		0.6	V
V_{IH}	Input High Voltage				2.0		V_{CC}+0.75	V
V_{OL}	Output Low Voltage V_{IN} = V_{IH} or V_{IL}	V_{CC} = 3.0 V	Com.,Ind./Mil.	I_{OL} = 8 mA / 6 mA			0.5	V
		V_{CC} = 4.5 V	Com.,Ind./Mil.	I_{OL} = 16 mA / 12 mA			0.5	V
		V_{CC} = 3.0 V	Com.,Ind./Mil.	I_{OL} = 6 mA / 4 mA			0.35	V
V_{OH}	Output High Voltage	V_{IN} = V_{IH} or V_{IL}, V_{CC} = 3.0 V / 4.5 V		I_{OH} = -100 µA	V_{CC}-0.3			V
				I_{OH} = -0.4 mA / -4.0 mA	2.4			V

Notes: 1. Not more than one output at a time should be shorted. Duration of short circuit test should not exceed 30 sec.
2. See I_{CC} vs. Frequency curves in the back of this data sheet.

A.C. Characteristics for the AT22LV10

Symbol	Parameter	AT22LV10-20			AT22LV10-25			AT22LV10-30			Units
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
t_{PD}	Input or Feedback to Non-Registered Output		12	20		15	25		20	30	ns
t_{EA}	Input to Output Enable			20		15	25		20	30	ns
t_{ER}	Input to Output Disable			20		15	25		20	30	ns
t_{CF}	Clock to Feedback	0	4	9	0	5	9	0	6	10	ns
t_{CO}	Clock to Output	0	8	14	0	10	17	0	12	20	ns
t_S	Input or Feedback Setup Time	10	6		12	7		15	8		ns
t_H	Hold Time	0			0			0			ns
t_P	Clock Period	10			12			14			ns
t_W	Clock Width	5			6			7			ns
F_{MAX}	External Feedback $1/(t_S+t_{CO})$			41.6			34.5			28.5	MHz
	Internal Feedback $1/(t_S+t_{CF})$			52.6			47.6			40.0	MHz
	No Feedback $1/(t_P)$			100.0			83.3			71.4	MHz
t_{AW}	Asynchronous Reset Width	20	12		25	15		30	18		ns
t_{AR}	Asynchronous Reset, Synchronous Preset, Recovery Time	20	12		25	15		30	18		ns
t_{AP}	Asynchronous Reset to Registered Output Reset		15	25		18	28		20	30	ns

A.C. Waveforms [1]

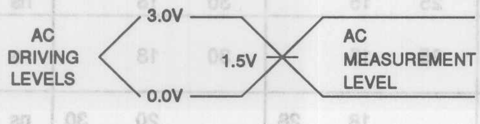


Note: 1. Timing measurement reference is 1.5 V. Input AC driving levels are 0.0 V and 3.0 V, unless otherwise specified.

A.C. Characteristics for the AT22LV10L

		AT22LV10L-25			AT22LV10L-30			AT22LV10L-35			
Symbol	Parameter	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Units
t_{PD}	Input or Feedback to Non-Registered Output		15	25		20	30		25	35	ns
t_{EA}	Input to Output Enable		15	25		20	30		25	35	ns
t_{ER}	Input to Output Disable		15	25		20	30		25	35	ns
t_{CF}	Clock to Feedback	0	5	9	0	6	10		7	11	ns
t_{CO}	Clock to Output	0	10	14	0	12	17		15	20	ns
t_{SF}	Feedback Setup Time	12	7		15	10		18	12		ns
t_S	Input Setup Time	17	15		20	15		22	15		ns
t_H	Hold Time	0			0			0			ns
t_P	Clock Period	12			14			16			ns
t_W	Clock Width	6			7			8			ns
F_{MAX}	External Feedback $1/(t_S+t_{CO})$			32.2			27.0			23.8	MHz
	Internal Feedback $1/(t_{SF}+t_{CF})$			47.6			40.0			34.4	MHz
	No Feedback $1/(t_P)$			83.3			71.4			62.5	MHz
t_{AW}	Asynchronous Reset Width	25	15		30	18		35	20		ns
t_{AR}	Asynchronous Reset Recovery Time	25	15		30	18		35	20		ns
t_{AP}	Asynchronous Reset to Registered Output Reset		18	28		20	30		22	35	ns

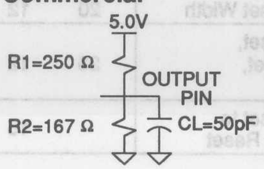
Input Test Waveforms and Measurement Levels



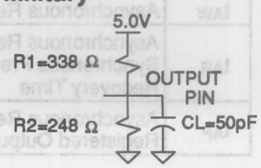
t_R, t_F < 5 ns (10% to 90%)

Output Test Loads:

Commercial

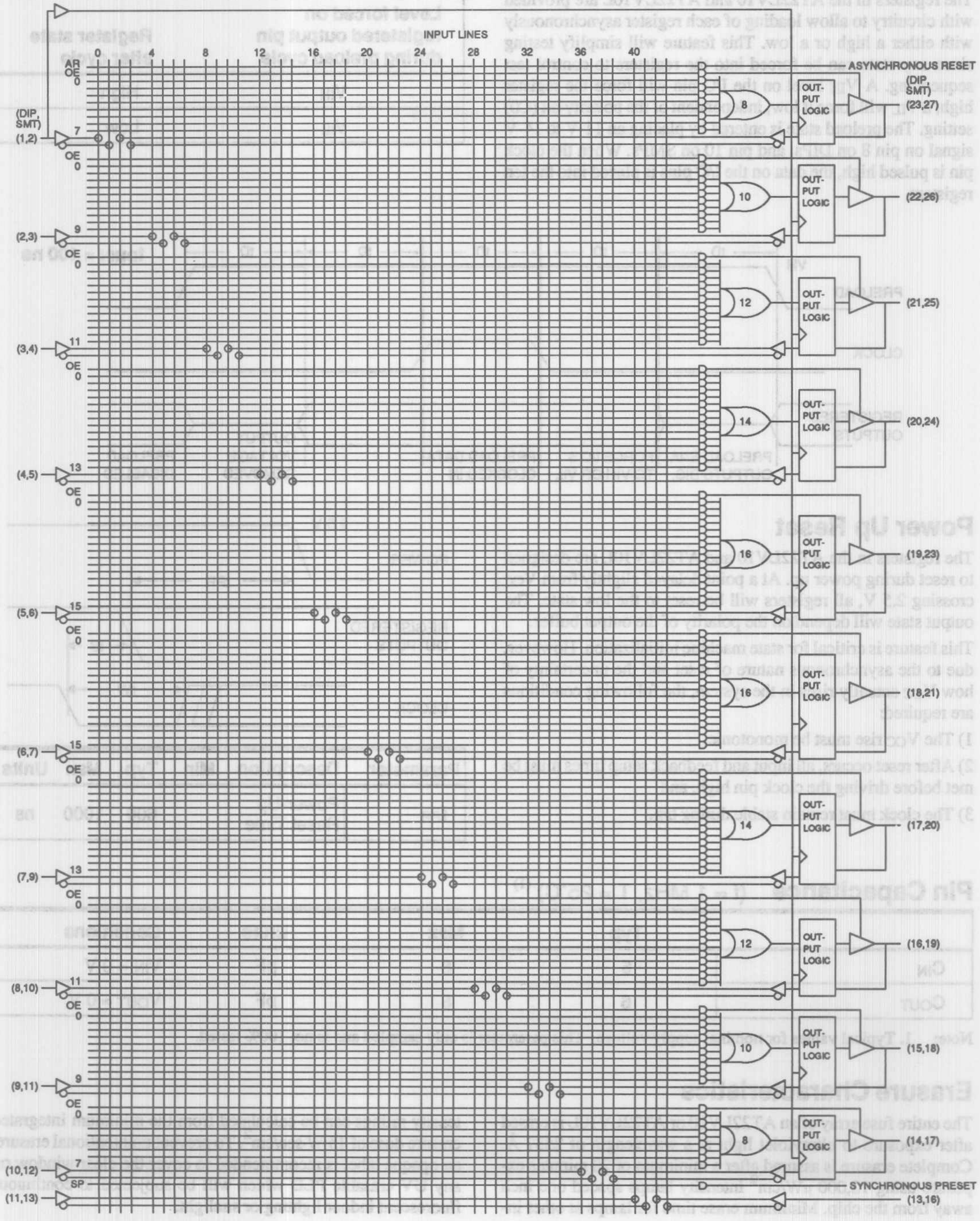


Military



1

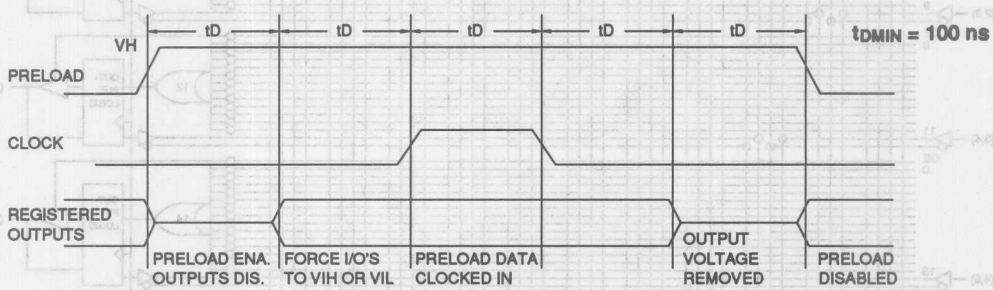
Functional Logic Diagram AT22LV10/L



Preload of Registered Outputs

The registers in the AT22LV10 and AT22LV10L are provided with circuitry to allow loading of each register asynchronously with either a high or a low. This feature will simplify testing since any state can be forced into the registers to control test sequencing. A V_{IH} level on the I/O pin will force the register high; a V_{IL} will force it low, independent of the polarity bit (C0) setting. The preload state is entered by placing an 11-V to 14-V signal on pin 8 on DIPs, and pin 10 on SMPs. When the clock pin is pulsed high, the data on the I/O pins is placed into the ten registers.

Level forced on registered output pin during preload cycle	Register state after cycle
V_{IH}	High
V_{IL}	Low



Power Up Reset

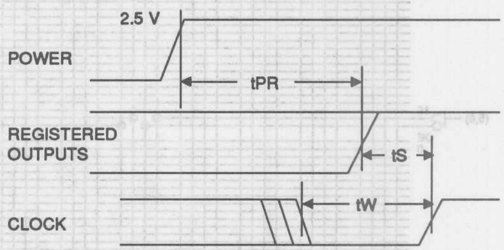
The registers in the AT22LV10 and AT22LV10L are designed to reset during power up. At a point delayed slightly from V_{CC} crossing 2.5 V, all registers will be reset to the low state. The output state will depend on the polarity of the output buffer.

This feature is critical for state machine initialization. However, due to the asynchronous nature of reset and the uncertainty of how V_{CC} actually rises in the system, the following conditions are required:

1) The V_{CC} rise must be monotonic,

2) After reset occurs, all input and feedback setup times must be met before driving the clock pin high, and

3) The clock must remain stable during t_{PR}.



Parameter	Description	Min	Typ	Max	Units
t_{PR}	Power-Up Reset Time		600	1000	ns

Pin Capacitance (f = 1 MHz, T = 25°C) [1]

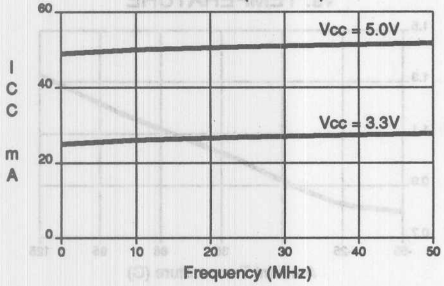
	Typ	Max	Units	Conditions
C_{IN}	5	8	pF	V_{IN} = 0 V
C_{OUT}	6	8	pF	V_{OUT} = 0 V

Note: 1. Typical values for nominal supply voltage. This parameter is only sampled and is not 100% tested.

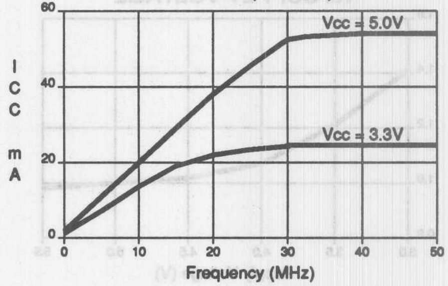
Erasure Characteristics

The entire fuse array of an AT22LV10 or AT22LV10L is erased after exposure to ultraviolet light at a wavelength of 2537 Å. Complete erasure is assured after a minimum of 20 minutes exposure using 12,000 µW/cm^2 intensity lamps spaced one inch away from the chip. Minimum erase time for lamps at other intensity ratings can be calculated from the minimum integrated erasure dose of 15 W·sec/cm^2. To prevent unintentional erasure, an opaque label is recommended to cover the clear window on any UV erasable PLD which will be subjected to continuous fluorescent indoor lighting or sunlight.

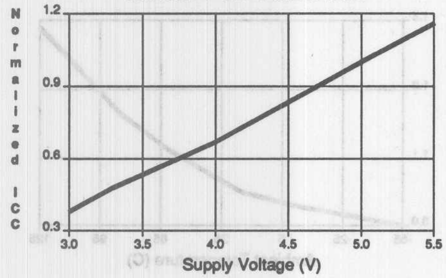
AT22LV10 (TA = 25C)



AT22LV10L (TA = 25C)

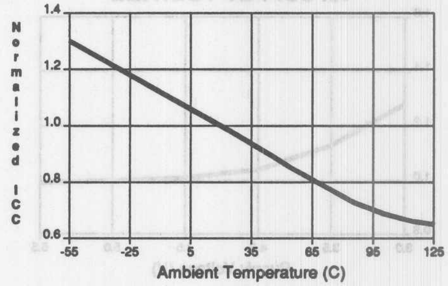


NORMALIZED SUPPLY CURRENT vs. SUPPLY VOLTAGE



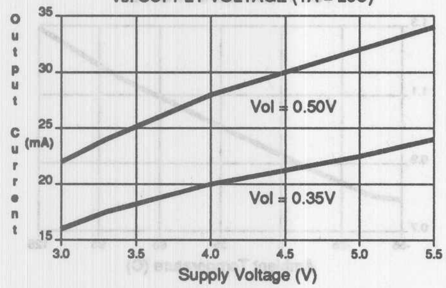
NORMALIZED ICC vs. AMBIENT TEMP.

f = 30 MHz



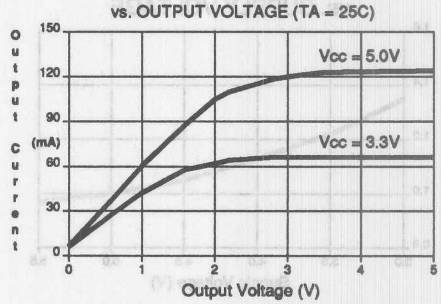
OUTPUT SINK CURRENT

vs. SUPPLY VOLTAGE (TA = 25C)



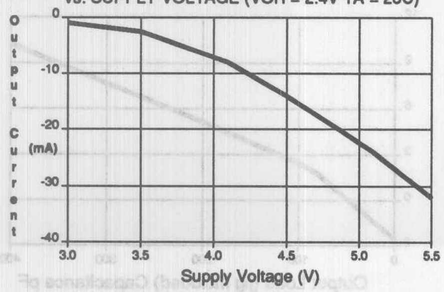
OUTPUT SINK CURRENT

vs. OUTPUT VOLTAGE (TA = 25C)



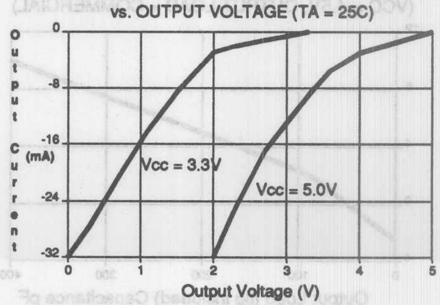
OUTPUT SOURCE CURRENT

vs. SUPPLY VOLTAGE (VOH = 2.4V TA = 25C)

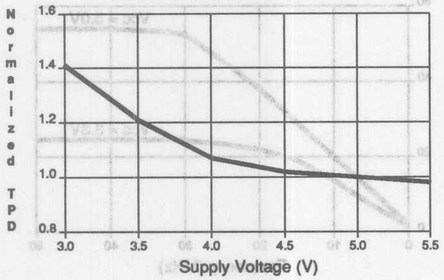


OUTPUT SOURCE CURRENT

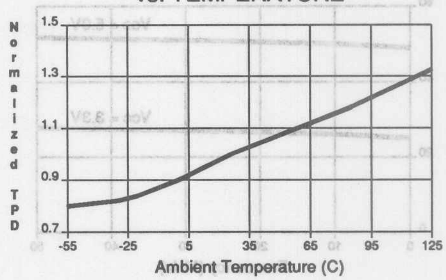
vs. OUTPUT VOLTAGE (TA = 25C)



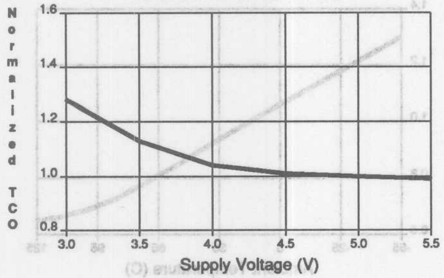
NORMALIZED TPD vs. SUPPLY VOLTAGE



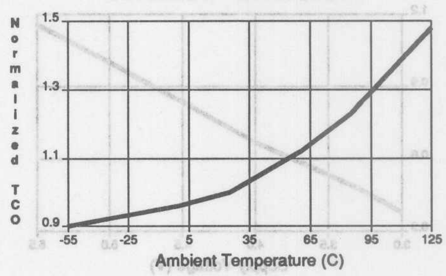
NORMALIZED TPD vs. TEMPERATURE



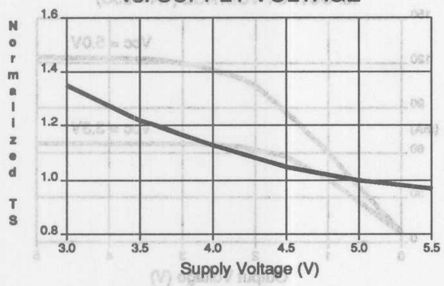
NORMALIZED TCO vs. SUPPLY VOLTAGE



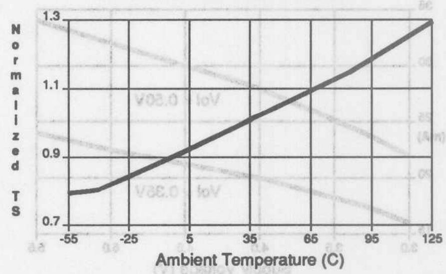
NORMALIZED TCO vs. TEMPERATURE



NORMALIZED TS vs. SUPPLY VOLTAGE

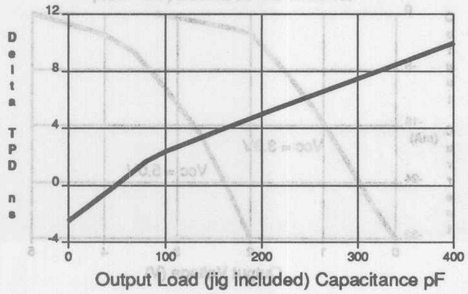


NORMALIZED TS vs. TEMPERATURE



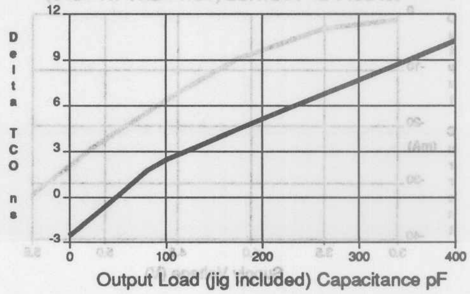
DELTA TPD vs. OUTPUT LOADING

(VCC = 4.5V, OUTPUT LOAD = COMMERCIAL)



DELTA TCO vs. OUTPUT LOADING

(VCC = 4.5V, OUTPUT LOAD = COMMERCIAL)



1

Ordering Information

tPD (ns)	ts (ns)	tCO (ns)	Ordering Code	Package	Operation Range
20	10	14	AT22LV10-20DC	24DW3	Commercial (0°C to 70°C)
			AT22LV10-20GC	24D3	
			AT22LV10-20JC	28J	
			AT22LV10-20KC	28KW	
			AT22LV10-20LC	28LW	
			AT22LV10-20NC	28L	
			AT22LV10-20PC	24P3	
			AT22LV10-20SC	24S	
			AT22LV10-20DI	24DW3	Industrial (-40°C to 85°C)
			AT22LV10-20GI	24D3	
			AT22LV10-20JI	28J	
			AT22LV10-20KI	28KW	
			AT22LV10-20LI	28LW	
			AT22LV10-20NI	28L	
			AT22LV10-20PI	24P3	
			AT22LV10-20SI	24S	
25	12	17	AT22LV10-25DC	24DW3	Commercial (0°C to 70°C)
			AT22LV10-25GC	24D3	
			AT22LV10-25JC	28J	
			AT22LV10-25KC	28KW	
			AT22LV10-25LC	28LW	
			AT22LV10-25NC	28L	
			AT22LV10-25PC	24P3	
			AT22LV10-25SC	24S	
			AT22LV10-25DI	24DW3	Industrial (-40°C to 85°C)
			AT22LV10-25GI	24D3	
			AT22LV10-25JI	28J	
			AT22LV10-25KI	28KW	
			AT22LV10-25LI	28LW	
			AT22LV10-25NI	28L	
			AT22LV10-25PI	24P3	
			AT22LV10-25SI	24S	
			AT22LV10-25DM	24DW3	Military (-55°C to 125°C)
			AT22LV10-25GM	24D3	
			AT22LV10-25KM	28KW	
			AT22LV10-25LM	28LW	
			AT22LV10-25NM	28L	
			AT22LV10-25DM/883	24DW3	Military/883D (-55°C to 125°C) Class B, Fully Compliant
			AT22LV10-25GM/883	24D3	
			AT22LV10-25KM/883	28KW	
			AT22LV10-25LM/883	28LW	
			AT22LV10-25NM/883	28L	
30	15	20	AT22LV10-30DC	24DW3	Commercial (0°C to 70°C)
			AT22LV10-30GC	24D3	
			AT22LV10-30JC	28J	
			AT22LV10-30KC	28KW	
			AT22LV10-30LC	28LW	
			AT22LV10-30NC	28L	
			AT22LV10-30PC	24P3	
			AT22LV10-30SC	24S	

Ordering Information

t_{PD} (ns)	t_S (ns)	t_{CO} (ns)	Ordering Code	Package	Operation Range
30	15	20	AT22LV10-30DI	24DW3	Industrial (-40°C to 85°C)
			AT22LV10-30GI	24D3	
			AT22LV10-30JI	28J	
			AT22LV10-30KI	28KW	
			AT22LV10-30LI	28LW	
			AT22LV10-30NI	28L	
			AT22LV10-30PI	24P3	
			AT22LV10-30SI	24S	
			AT22LV10-30DM	24DW3	Military (-55°C to 125°C)
			AT22LV10-30GM	24D3	
			AT22LV10-30KM	28KW	
			AT22LV10-30LM	28LW	
			AT22LV10-30NM	28L	
			AT22LV10-30DM/883	24DW3	Military/883D (-55°C to 125°C) Class B, Fully Compliant
			AT22LV10-30GM/883	24D3	
			AT22LV10-30KM/883	28KW	
			AT22LV10-30LM/883	28LW	
			AT22LV10-30NM/883	28L	

Ordering Information

t_{PD} (ns)	t_S (ns)	t_{CO} (ns)	Ordering Code	Package	Operation Range
25	17	14	AT22LV10L-25DC AT22LV10L-25GC AT22LV10L-25JC AT22LV10L-25KC AT22LV10L-25LC AT22LV10L-25NC AT22LV10L-25PC AT22LV10L-25SC	24DW3 24D3 28J 28KW 28LW 28L 24P3 24S	Commercial (0°C to 70°C)
			AT22LV10L-25DI AT22LV10L-25GI AT22LV10L-25JI AT22LV10L-25KI AT22LV10L-25LI AT22LV10L-25NI AT22LV10L-25PI AT22LV10L-25SI	24DW3 24D3 28J 28KW 28LW 28L 24P3 24S	Industrial (-40°C to 85°C)
30	20	17	AT22LV10L-30DC AT22LV10L-30GC AT22LV10L-30JC AT22LV10L-30KC AT22LV10L-30LC AT22LV10L-30NC AT22LV10L-30PC AT22LV10L-30SC	24DW3 24D3 28J 28KW 28LW 28L 24P3 24S	Commercial (0°C to 70°C)
			AT22LV10L-30DI AT22LV10L-30GI AT22LV10L-30JI AT22LV10L-30KI AT22LV10L-30LI AT22LV10L-30NI AT22LV10L-30PI AT22LV10L-30SI	24DW3 24D3 28J 28KW 28LW 28L 24P3 24S	Industrial (-40°C to 85°C)
			AT22LV10L-30DM AT22LV10L-30GM AT22LV10L-30KM AT22LV10L-30LM AT22LV10L-30NM	24DW3 24D3 28KW 28LW 28L	Military (-55°C to 125°C)
			AT22LV10L-30DM/883 AT22LV10L-30GM/883 AT22LV10L-30KM/883 AT22LV10L-30LM/883 AT22LV10L-30NM/883	24DW3 24D3 28KW 28LW 28L	Military/883D (-55°C to 125°C) Class B, Fully Compliant
35	22	20	AT22LV10L-35DC AT22LV10L-35GC AT22LV10L-35JC AT22LV10L-35KC AT22LV10L-35LC AT22LV10L-35NC AT22LV10L-35PC AT22LV10L-35SC	24DW3 24D3 28J 28KW 28LW 28L 24P3 24S	Commercial (0°C to 70°C)

Ordering Information

tPD (ns)	ts (ns)	tCO (ns)	Ordering Code	Package	Operation Range
35	22	20	AT22LV10L-35DI AT22LV10L-35GI AT22LV10L-35JI AT22LV10L-35KI AT22LV10L-35LI AT22LV10L-35NI AT22LV10L-35PI AT22LV10L-35SI	24DW3 24D3 28J 28KW 28LW 28L 24P3 24S	Industrial (-40°C to 85°C)
			AT22LV10L-35DM AT22LV10L-35GM AT22LV10L-35KM AT22LV10L-35LM AT22LV10L-35NM	24DW3 24D3 28KW 28LW 28L	Military (-55°C to 125°C)
			AT22LV10L-35DM/883 AT22LV10L-35GM/883 AT22LV10L-35KM/883 AT22LV10L-35LM/883 AT22LV10L-35NM/883	24DW3 24D3 28KW 28LW 28L	Military/883D (-55°C to 125°C) Class B, Fully Compliant

Package Type	
24DW3	24 Lead, 0.300" Wide, Windowed, Ceramic Dual Inline Package (Cerdip)
24D3	24 Lead, 0.300" Wide, Non-Windowed (OTP), Ceramic Dual Inline Package (Cerdip)
28J	28 Lead, Plastic J-Leaded Chip Carrier OTP (PLCC)
28KW	28 Lead, Windowed, Ceramic J-Leaded Chip Carrier (JLCC)
28LW	28 Pad, Windowed, Ceramic Leadless Chip Carrier (LCC)
28L	28 Pad, Non-Windowed, Ceramic Leadless Chip Carrier OTP (LCC)
24P3	24 Lead, 0.300" Wide, Plastic Dual Inline Package OTP (PDIP)
24S	24 Lead, 0.300" Wide, Plastic Gull Wing Small Outline OTP (SOIC)

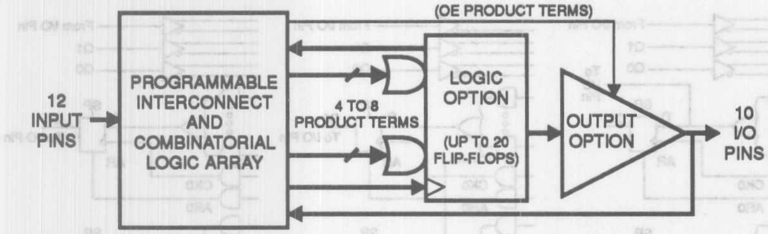
1

High Density UV Erasable Programmable Logic Device

Features

- Third Generation Programmable Logic Structure
  High Density Replacement for Discrete Logic
- High Speed - Plus a New Low Power Version
- Increased Logic Flexibility
  42 Inputs and 20 Sum terms
- Flexible Output Logic
  20 Flip-Flops - 10 Extra
  All Can Be Individually Buried or 10 Output Directly
  Each has Individual Asynchronous Reset and Clock Terms
- Multiple Feedback Paths Provide for Buried State Machines and I/O Bus Compatibility
- Proven and Reliable High Speed CMOS EPROM Process
  2000 V ESD Protection
  200 mA Latchup Immunity
- Reprogrammable - Tested 100% for Programmability
- 24-pin, 300-mil Dual-In-line and 28-Lead Surface Mount Packages

Logic Diagram



Description

The ATV750/L is 100% more powerful than most other programmable logic devices in 24-pin packages. Increased product terms, sum terms, and flip-flops translate into more usable gates.

Each of the ATV750's 22 logic pins can be used as an input. Ten of these can be used as input, output, or bi-directional I/O pins. All 20 flip-flops can be fed back into the array independently. This flexibility allows burying all of the sum terms and flip-flops.

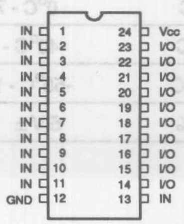
There are 171 product terms available. A variable format is used to assign between four and eight product terms per sum term. There are two sum terms per output, providing added flexibility.

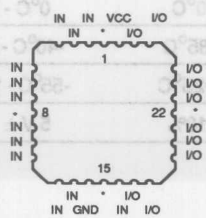
The ATV750/L has more flip-flops available than other PLDs in this density range. Complex state machines are easily implemented.

Product terms are available providing asynchronous resets, flip-flop clocks, and output enables. One reset and one clock term are provided per flip-flop, with one enable term per output. One product term provides a global synchronous preset. Register preload simplifies testing. The device has an internal power up clear function.

Pin Configurations

Pin Name	Function
IN	Logic Inputs
I/O	Bidirectional Buffers
*	No Internal Connection
Vcc	+5V Supply





Absolute Maximum Ratings*

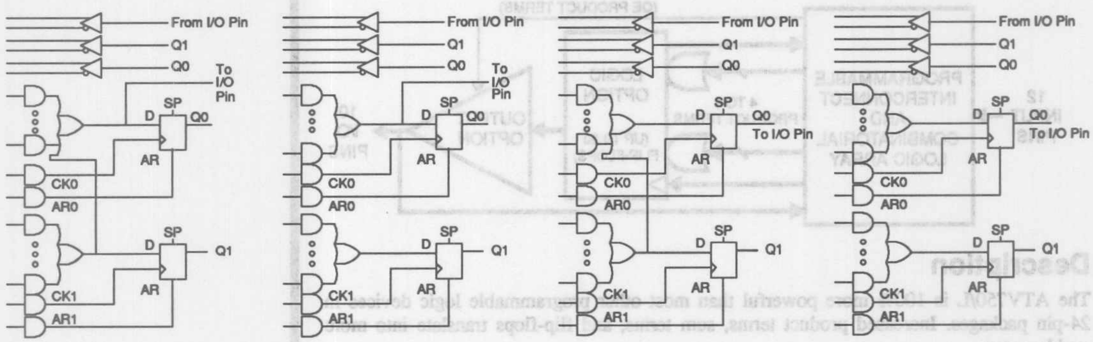
Temperature Under Bias	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-2.0 V to +7.0 V(1)
Voltage on Input Pins with Respect to Ground During Programming	-2.0 V to +14.0 V(1)
Programming Voltage with Respect to Ground	-2.0 V to +14.0 V(1)
Integrated UV Erase Dose	7258 W· sec/cm²

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

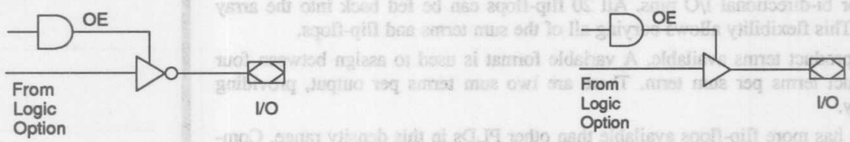
Note:
1. Minimum voltage is -0.6 V dc which may undershoot to -2.0 V for pulses of less than 20 ns. Maximum output pin voltage is V_{CC}+0.75 V dc which may overshoot to +7.0 V for pulses of less than 20 ns.

Logic Options

Combined Terms | Separate Terms | Combined Terms | Separate Terms



Output Options



D.C. and A.C. Operating Conditions

		ATV750-20	ATV750/L-25	ATV750/L-30	ATV750-35
Operating Temperature (Case)	Com.	0°C - 70°C	0°C - 70°C	0°C - 70°C	0°C - 70°C
	Ind.	-40°C - 85°C	-40°C - 85°C	-40°C - 85°C	-40°C - 85°C
	Mil.	-55°C - 125°C	-55°C - 125°C	-55°C - 125°C	-55°C - 125°C
V_{CC} Power Supply		5 V ± 10%	5 V ± 10%	5 V ± 10%	5 V ± 10%

D.C. Characteristics

Symbol	Parameter	Condition			Min	Typ	Max	Units
I_{LI}	Input Load Current	V_{IN} = -0.1 V to V_{CC}+1 V					10	µA
I_{LO}	Output Leakage Current	V_{OUT} = -0.1 V to V_{CC}+0.1 V					10	µA
I_{CC}	Power Supply Current	VCC = MAX, V_{IN} = GND, Outputs Open	ATV750	Com.			120	mA
				Ind.,Mil.			140	mA
			ATV750L	Com.		1.0	12	mA
				Ind.,Mil.		1.0	15	mA
I_{CC2} (2)	Clocked Power Supply Current	f = 1MHz, VCC = MAX, Outputs Open	ATV750L	Com.			15	mA
				Ind.,Mil.			20	mA
I_{OS} (1)	Output Short Circuit Current	V_{OUT} = 0.5 V					-90	mA
V_{IL}	Input Low Voltage				-0.6		0.8	V
V_{IH}	Input High Voltage				2.0		V_{CC}+0.75	V
V_{OL}	Output Low Voltage	V_{IN} = V_{IH} or V_{IL}, V_{CC} = MIN	I_{OL} = 12 mA Com.,Ind.				0.5	V
			I_{OL} = 8 mA Mil.				0.5	V
			I_{OL} = 24 mA, Com.				1.0	V
V_{OH}	Output High Voltage	V_{IN} = V_{IH} or V_{IL}, V_{CC} = MIN	I_{OH} = -100 µA		V_{CC}-0.3			V
			I_{OH} = -4.0 mA		2.4			V

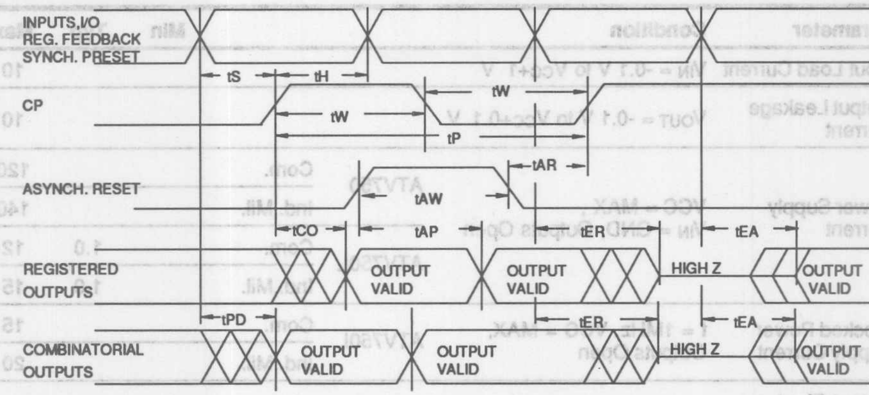
Notes: 1. Not more than one output at a time should be shorted. Duration of short circuit test should not exceed 30 sec.
2. Outputs not loaded.

Operating Modes

Mode	24-Pin DIP	1	5	8	11	13	I/Os	Vcc (24)
	28-Pin JLCC	2	6	10	13	16	I/Os	Vcc (28)
"PLD"		X (1)	X	X	X	X	I/O	5 V
Program		V_{PP}	X/V_H (2)	X	X/V_H	V_{PP}	D_{IN}	6 V
PGM Verify		V_{PP}	X/V_H	X	X/V_H	V_{IL}	D_{OUT}	5 V
PGM Inhibit		V_{PP}	X/V_H	X	X/V_H	V_{IH}	High Z	5-6V
Preload #1		X	X	V_H	X	V_{IL}	D_{IN}	5 V
Preload #2		X	X	V_H	X	V_{IH}	D_{IN}	5 V

Notes: 1. X can be V_{IL} or V_{IH}.
2. V_H = 11.0 V to 14.0 V

A.C. Waveforms [1]



Note: 1. Timing measurement reference is 1.5 V. Input AC driving levels are 0.0 V and 3.0 V, unless otherwise specified.

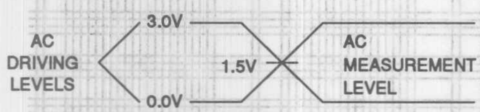
A.C. Characteristics

Symbol	Parameter	ATV750-30 Min	ATV750-30 Max	ATV750-35 Min	ATV750-35 Max	Units
t_{PD}	Input or Feedback to Non-Registered Output		30		35	ns
t_{EA}	Input to Output Enable		30		35	ns
t_{ER}	Input to Output Disable		30		35	ns
t_{CO}	Clock to Output	5	25	10	30	ns
t_{CF}	Clock to Feedback	5	10	10	12	ns
t_S	Input Setup Time	15		18		ns
t_H	Hold Time	5		10		ns
t_P	Clock Period	25		30		ns
t_W	Clock Width	12		15		ns
F_{MAX}	Maximum Frequency		40		33	MHz
t_{AW}	Asynchronous Reset Width	30		35		ns
t_{AR}	Asynchronous Reset Recovery Time	30		35		ns
t_{AP}	Asynchronous Reset to Registered Output Reset		30		35	ns
t_{SP}	Setup Time, Synchronous Preset	15		18		ns

A.C. Characteristics

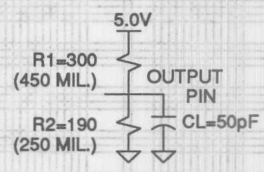
Symbol	Parameter	ATV750-20 Min	ATV750-20 Max	ATV750/L-25 Min	ATV750/L-25 Max	ATV750L-30 Min	ATV750L-30 Max	Units
t_{PD}	Input or Feedback to Non-Registered Output		20		25		30	ns
t_{EA}	Input to Output Enable		20		25		30	ns
t_{ER}	Input to Output Disable		20		25		30	ns
t_{CO}	Clock to Output		20		22	5	25	ns
t_{CF}	Clock to Feedback	5	10	5	10	5	10	ns
t_S	Input Setup Time	10		12		15		ns
t_{SF}	Feedback Setup Time	5		7		15		ns
t_H	Hold Time	5		5		5		ns
t_P	Clock Period	18		22		25		ns
t_W	Clock Width	8		10		12		ns
F_{MAX}	Maximum Frequency		55		45		40	MHz
t_{AW}	Asynchronous Reset Width	15		20		30		ns
t_{AR}	Asynchronous Reset Recovery Time	15		20		30		ns
t_{AP}	Asynchronous Reset to Registered Output Reset		20		25		30	ns
t_{SP}	Setup Time, Synchronous Preset	12		15		15		ns

Input Test Waveforms and Measurement Levels

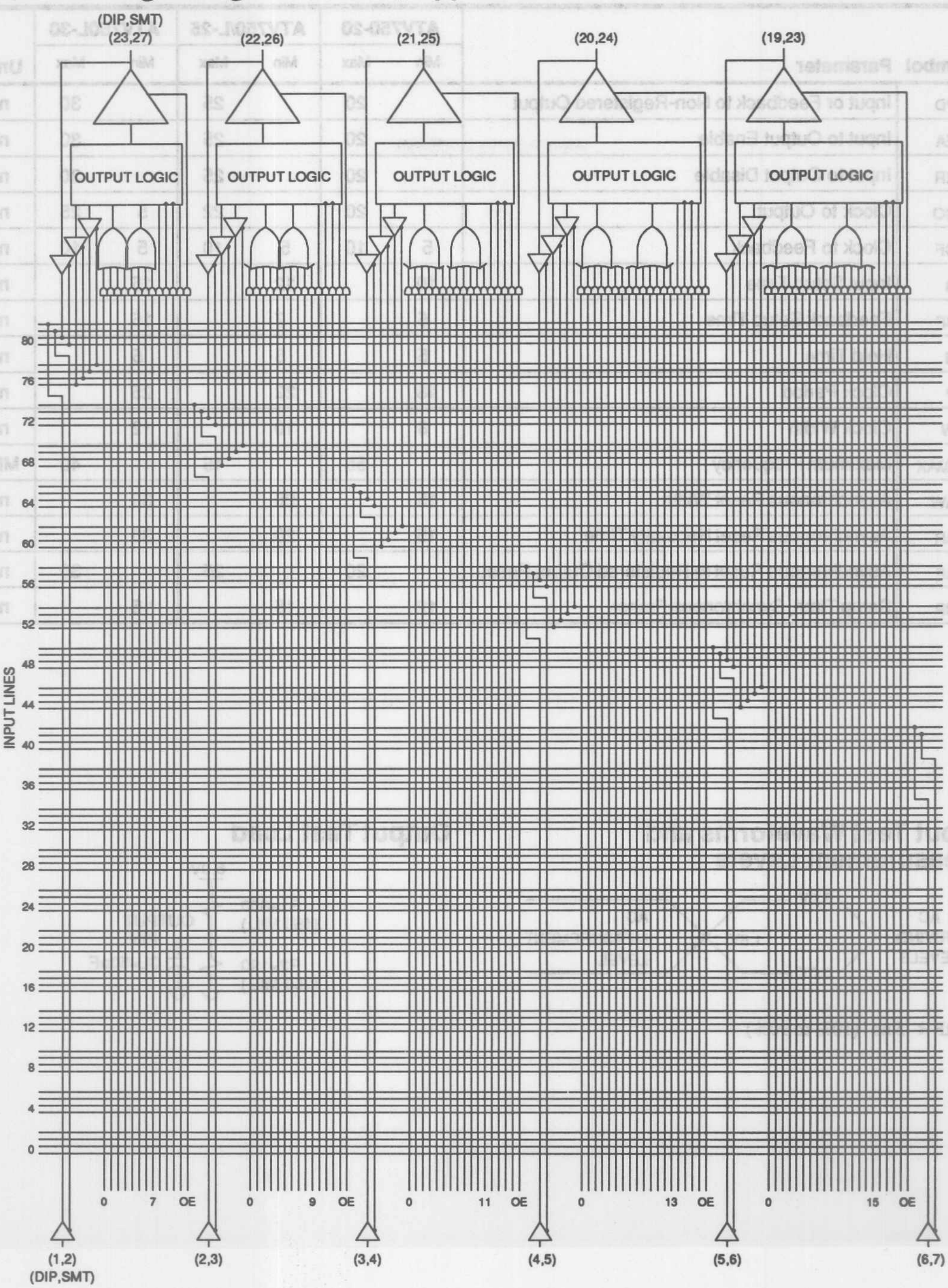


$t_R, t_F < 5$ ns (10% to 90%)

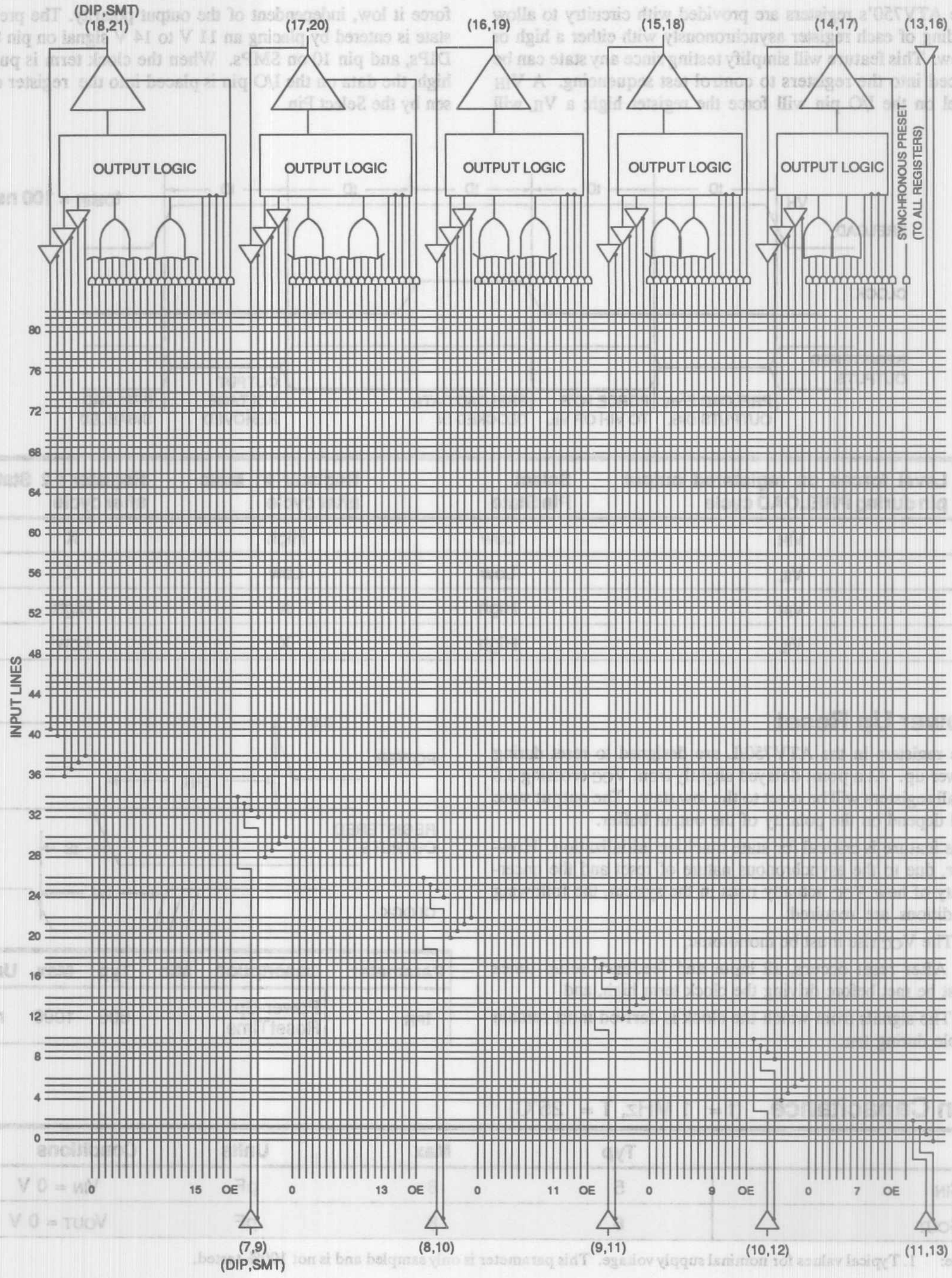
Output Test Load



Functional Logic Diagram ATV750, Upper Half

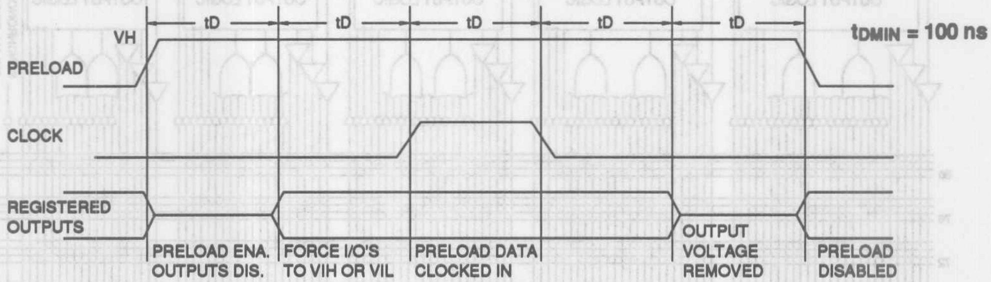


Functional Logic Diagram ATV750, Lower Half



Preload of Registered Outputs

The ATV750's registers are provided with circuitry to allow loading of each register asynchronously with either a high or a low. This feature will simplify testing since any state can be forced into the registers to control test sequencing. A V_{IH} level on the I/O pin will force the register high; a V_{IL} will force it low, independent of the output polarity. The preload state is entered by placing an 11 V to 14 V signal on pin 8 on DIPs, and pin 10 on SMPs. When the clock term is pulsed high, the data on the I/O pin is placed into the register chosen by the Select Pin.



Level forced on registered output pin during PRELOAD cycle	Select Pin State	Register #1 state after cycle	Register # 2 State after cycle
V_{IH}	Low	High	X
V_{IL}	Low	Low	X
V_{IH}	High	X	High
V_{IL}	High	X	Low

Power Up Reset

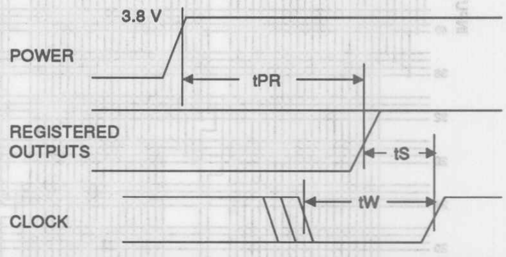
The registers in the ATV750/L are designed to reset during power up. At a point delayed slightly from V_{CC} crossing 3.8 V, all registers will be reset to the low state. The output state will depend on the polarity of the output buffer.

This feature is critical for state machine initialization. However, due to the asynchronous nature of reset and the uncertainty of how V_{CC} actually rises in the system, the following conditions are required:

1) The V_{CC} rise must be monotonic,

2) After reset occurs, all input and feedback setup times must be met before driving the clock term high, and

3) The signals from which the clock is derived must remain stable during t_{PR}.



Parameter	Description	Min	Typ	Max	Units
t_{PR}	Power-Up Reset Time		600	1000	ns

Pin Capacitance (f = 1 MHz, T = 25°C) [1]

	Typ	Max	Units	Conditions
C_{IN}	5	8	pF	V_{IN} = 0 V
C_{OUT}	6	8	pF	V_{OUT} = 0 V

Note: 1. Typical values for nominal supply voltage. This parameter is only sampled and is not 100% tested.

Using the ATV750's Many Advanced Features

The ATV750's flexibility puts more usable gates in 24 pins than other PLDs. The ATV750/L starts with an architecture similar to the popular AT22V10, and adds several features:

- Asynchronous Clocks -
  Each of the flip-flops in the ATV750/L has a dedicated product term driving the clock. The user is no longer constrained to using one clock for all the registers. Buried state machines, counters, and registers can all coexist in one device, while running on separate clocks. The ATV750/L clock period matches that of similar synchronous devices.
- A Full Bank of 10 More Registers -
  The ATV750/L provides two flip-flops for each output macrocell - a total of 20. Each register has its own clock and reset product terms, as well as its own SUM term.
- Independent I/O Pin and Feedback Paths -
  Each I/O pin on the ATV750/L has a dedicated input path. Each of the 20 registers has individual feedback terms into the array. This feature, combined with individual product terms for each I/O's output enable, facilitates designs using bi-directional I/O buses.
- Combinable Sum Terms -
  Each output macrocell's two SUM terms can be combined in an OR gate before the output or the register. This provides up to 16 product terms per output or flip-flop. This architecture increases the number of usable gates available.

Programming Software Support

Software which is capable of transforming Boolean equations, state machine descriptions and truth tables into JEDEC files for the ATV750/L is available from the following sources:

Data I/O Corp. - Abel 2.1, 3.0, and above
Logical Devices - Cupl 2.15B, and above

Synchronous Preset and Asynchronous Reset

One synchronous preset line is provided for all 20 registers in the ATV750/L. The appropriate input signals to cause the internal clocks to go to a high state must be received during a synchronous preset. Appropriate setup and hold times must be met, as shown in the switching waveform diagram.

An individual asynchronous reset line is provided for each of the 20 flip-flops. Both master and slave halves of the flip-flops are reset when the input signals received combine so as to force the internal resets high.

Security Fuse Usage

A single fuse is provided to prevent unauthorized copying of the ATV750/L fuse patterns. Once programmed, the output buffers will remain in a high impedance state during verify.

The security fuse should be programmed last, as its effect is immediate.

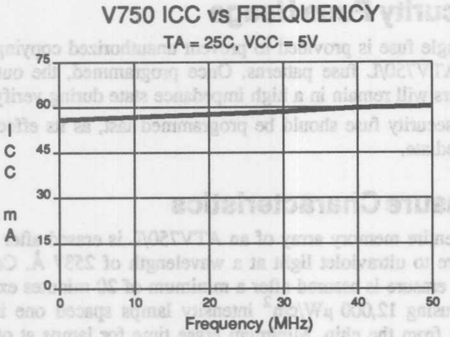
Erasure Characteristics

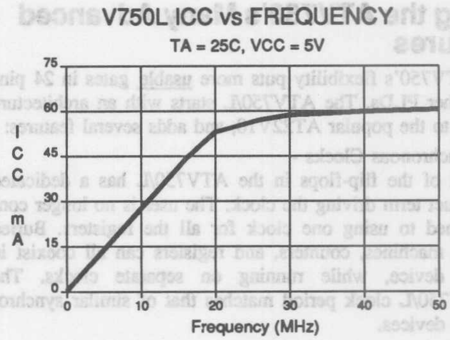
The entire memory array of an ATV750/L is erased after exposure to ultraviolet light at a wavelength of 2537 Å. Complete erasure is assured after a minimum of 20 minutes exposure using 12,000 $\mu W/cm^2$ intensity lamps spaced one inch away from the chip. Minimum erase time for lamps at other intensity ratings can be calculated from the minimum integrated erasure dose of 15 W· sec/cm^2. To prevent unintentional erasure, an opaque label is recommended to cover the clear window on any UV erasable PLD which will be subjected to continuous fluorescent indoor lighting or sunlight.

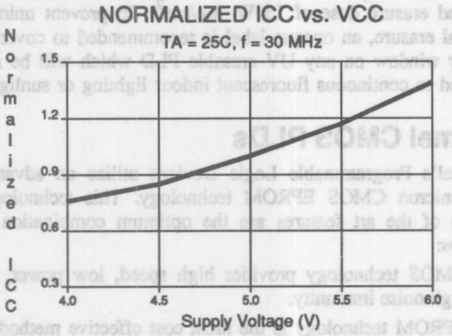
Atmel CMOS PLDs

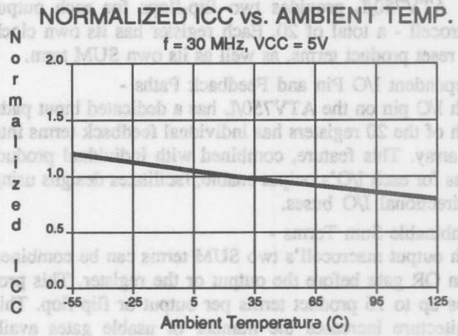
Atmel's Programmable Logic Devices utilize an advanced 1.5-micron CMOS EPROM technology. This technology's state of the art features are the optimum combination for PLDs:

- CMOS technology provides high speed, low power, and high noise immunity.
- EPROM technology is the most cost effective method for producing PLDs - surpassing bipolar fusible link technology in low cost, while providing the necessary reprogrammability.
- EPROM reprogrammability, which is 100% tested before shipment, provides inherently better programmability and reliability than one-time fusible PLDs.
- Atmel's EPROM process has proven extremely reliable in the volume production of a full line of advanced EPROM memory products, from 64K to one-megabit devices.


V750 ICC vs FREQUENCY
TA = 25C, VCC = 5V
75
60
45
30
15
0
ICC mA
0
10
20
30
40
50
Frequency (MHz)


V750L ICC vs FREQUENCY
TA = 25C, VCC = 5V
75
60
45
30
15
0
ICC mA
0
10
20
30
40
50
Frequency (MHz)


NORMALIZED ICC vs. VCC
TA = 25C, f = 30 MHz
1.5
1.2
0.9
0.6
0.3
Normalized ICC
4.0
4.5
5.0
5.5
6.0
Supply Voltage (V)


NORMALIZED ICC vs. AMBIENT TEMP.
f = 30 MHz, VCC = 5V
2.0
1.5
1.0
0.5
0.0
Normalized ICC
-55
-25
5
35
65
95
125
Ambient Temperature (C)

Ordering Information

t_{PD} (ns)	t_{CO} (ns)	f_{MAX} (MHz)	Ordering Code	Package	Operation Range
20	20	55	ATV750-20DC	24DW3	Commercial (0°C to 70°C)
			ATV750-20GC	24D3	
			ATV750-20JC	28J	
			ATV750-20KC	28KW	
			ATV750-20LC	28LW	
			ATV750-20NC	28L	
			ATV750-20PC	24P3	
			ATV750-20SC	24S	
			ATV750-20DI	24DW3	Industrial (-40°C to 85°C)
			ATV750-20GI	24D3	
			ATV750-20JI	28J	
			ATV750-20KI	28KW	
			ATV750-20LI	28LW	
			ATV750-20NI	28L	
			ATV750-20PI	24P3	
			ATV750-20SI	24S	
			ATV750-20DM	24DW3	Military (-55°C to 125°C)
			ATV750-20GM	24D3	
			ATV750-20KM	28KW	
			ATV750-20LM	28LW	
			ATV750-20NM	28L	
			ATV750-20DM/883	24DW3	Military/883C (-55°C to 125°C) Class B, Fully Compliant
			ATV750-20GM/883	24D3	
			ATV750-20KM/883	28KW	
			ATV750-20LM/883	28LW	
			ATV750-20NM/883	28L	
25	22	45	ATV750-25DC	24DW3	Commercial (0°C to 70°C)
			ATV750-25GC	24D3	
			ATV750-25JC	28J	
			ATV750-25KC	28KW	
			ATV750-25LC	28LW	
			ATV750-25NC	28L	
			ATV750-25PC	24P3	
			ATV750-25SC	24S	
			ATV750-25DI	24DW3	Industrial (-40°C to 85°C)
			ATV750-25GI	24D3	
			ATV750-25JI	28J	
			ATV750-25KI	28KW	
			ATV750-25LI	28LW	
			ATV750-25NI	28L	
			ATV750-25PI	24P3	
			ATV750-25SI	24S	

Ordering Information

t_{PD} (ns)	t_{CO} (ns)	f_{MAX} (MHz)	Ordering Code	Package	Operation Range
25	22	45	ATV750-25DM ATV750-25GM ATV750-25KM ATV750-25LM ATV750-25NM	24DW3 24D3 28KW 28LW 28L	Military (-55°C to 125°C)
			ATV750-25DM/883 ATV750-25GM/883 ATV750-25KM/883 ATV750-25LM/883 ATV750-25NM/883	24DW3 24D3 28KW 28LW 28L	Military/883C (-55°C to 125°C) Class B, Fully Compliant
30	25	40	ATV750-30DC ATV750-30GC ATV750-30JC ATV750-30KC ATV750-30LC ATV750-30NC ATV750-30PC ATV750-30SC	24DW3 24D3 28J 28KW 28LW 28L 24P3 24S	Commercial (0°C to 70°C)
			ATV750-30DI ATV750-30GI ATV750-30JI ATV750-30KI ATV750-30LI ATV750-30NI ATV750-30PI ATV750-30SI	24DW3 24D3 28J 28KW 28LW 28L 24P3 24S	Industrial (-40°C to 85°C)
			ATV750-30DM ATV750-30GM ATV750-30KM ATV750-30LM ATV750-30NM	24DW3 24D3 28KW 28LW 28L	Military (-55°C to 125°C)
			ATV750-30DM/883 ATV750-30GM/883 ATV750-30KM/883 ATV750-30LM/883 ATV750-30NM/883	24DW3 24D3 28KW 28LW 28L	Military/883C (-55°C to 125°C) Class B, Fully Compliant
35	30	33	ATV750-35DC ATV750-35GC ATV750-35JC ATV750-35KC ATV750-35LC ATV750-35NC ATV750-35PC ATV750-35SC	24DW3 24D3 28J 28KW 28LW 28L 24P3 24S	Commercial (0°C to 70°C)

Ordering Information

t_{PD} (ns)	t_{CO} (ns)	f_{MAX} (MHz)	Ordering Code	Package	Operation Range
35	30	33	ATV750-35DI ATV750-35GI ATV750-35JI ATV750-35KI ATV750-35LI ATV750-35NI ATV750-35PI ATV750-35SI	24DW3 24D3 28J 28KW 28LW 28L 24P3 24S	Industrial (-40°C to 85°C)
			ATV750-35DM ATV750-35GM ATV750-35KM ATV750-35LM ATV750-35NM	24DW3 24D3 28KW 28LW 28L	Military (-55°C to 125°C)
			ATV750-35DM/883 ATV750-35GM/883 ATV750-35KM/883 ATV750-35LM/883 ATV750-35NM/883	24DW3 24D3 28KW 28LW 28L	Military/883C (-55°C to 125°C) Class B, Fully Compliant
20	20	55	5962-8872604LX 5962-88726043X	24DW3 28LW	Military/883C (-55°C to 125°C) Class B, Fully Compliant
25	22	45	5962-8872603LX 5962-88726033X	24DW3 28LW	Military/883C (-55°C to 125°C) Class B, Fully Compliant
35	30	33	5962-8872602LX 5962-88726023X	24DW3 28LW	Military/883C (-55°C to 125°C) Class B, Fully Compliant

Ordering Information

t_{PD} (ns)	t_{CO} (ns)	f_{MAX} (MHz)	Ordering Code	Package	Operation Range
25	22	45	ATV750L-25DC	24DW3	Commercial (0°C to 70°C)
			ATV750L-25GC	24D3	
			ATV750L-25JC	28J	
			ATV750L-25KC	28KW	
			ATV750L-25LC	28LW	
			ATV750L-25NC	28L	
			ATV750L-25PC	24P3	
			ATV750L-25SC	24S	
			ATV750L-25DI	24DW3	Industrial (-40°C to 85°C)
			ATV750L-25GI	24D3	
			ATV750L-25JI	28J	
			ATV750L-25KI	28KW	
			ATV750L-25LI	28LW	
			ATV750L-25NI	28L	
			ATV750L-25PI	24P3	
			ATV750L-25SI	24S	
			ATV750L-25DM	24DW3	Military (-55°C to 125°C)
			ATV750L-25GM	24D3	
			ATV750L-25KM	28KW	
			ATV750L-25LM	28LW	
			ATV750L-25NM	28L	
			ATV750L-25DM/883	24DW3	Military/883C (-55°C to 125°C) Class B, Fully Compliant
			ATV750L-25GM/883	24D3	
			ATV750L-25KM/883	28KW	
			ATV750L-25LM/883	28LW	
			ATV750L-25NM/883	28L	
30	25	40	ATV750L-30DC	24DW3	Commercial (0°C to 70°C)
			ATV750L-30GC	24D3	
			ATV750L-30JC	28J	
			ATV750L-30KC	28KW	
			ATV750L-30LC	28LW	
			ATV750L-30NC	28L	
			ATV750L-30PC	24P3	
			ATV750L-30SC	24S	
			ATV750L-30DI	24DW3	Industrial (-40°C to 85°C)
			ATV750L-30GI	24D3	
			ATV750L-30JI	28J	
			ATV750L-30KI	28KW	
			ATV750L-30LI	28LW	
			ATV750L-30NI	28L	
			ATV750L-30PI	24P3	
			ATV750L-30SI	24S	

Ordering Information

t_{PD} (ns)	t_{CO} (ns)	f_{MAX} (MHz)	Ordering Code	Package	Operation Range
30	25	40	ATV750L-30DM ATV750L-30GM ATV750L-30KM ATV750L-30LM ATV750L-30NM	24DW3 24D3 28KW 28LW 28L	Military (-55°C to 125°C)
			ATV750L-30DM/883 ATV750L-30GM/883 ATV750L-30KM/883 ATV750L-30LM/883 ATV750L-30NM/883	24DW3 24D3 28KW 28LW 28L	Military/883C (-55°C to 125°C) Class B, Fully Compliant
25	22	45	5962-8872606LX 5962-88726063X	24DW3 28LW	Military/883C (-55°C to 125°C) Class B, Fully Compliant
30	25	40	5962-8872606LX 5962-88726063X	24DW3 28LW	Military/883C (-55°C to 125°C) Class B, Fully Compliant

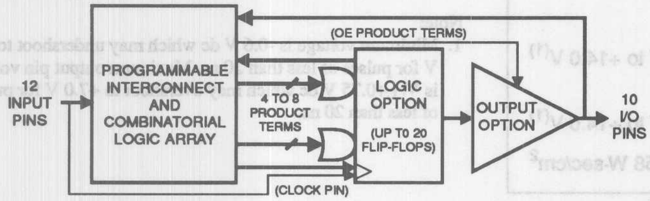
Package Type	
24DW3	24 Lead, 0.300" Wide, Windowed, Ceramic Dual Inline Package (Cerdip)
24D3	24 Lead, 0.300" Wide, Non-Windowed (OTP) Ceramic Dual Inline Package (Cerdip)
28J	28 Lead, Plastic J-Leaded Chip Carrier OTP (PLCC)
28KW	28 Lead, Windowed, Ceramic J-Leaded Chip Carrier (JLCC)
28LW	28 Pad, Windowed, Ceramic Leadless Chip Carrier (LCC)
28L	28 Pad, Non-Windowed, Ceramic Leadless Chip Carrier OTP (LCC)
24P3	24 Lead, 0.300" Wide, Plastic Dual Inline Package OTP (PDIP)
24S	24 Lead, 0.300" Wide, Plastic Gull Wing Small Outline OTP (SOIC)

High Speed UV Erasable Programmable Logic Device

Features

- Advanced, High Speed Programmable Logic Device
  - Improved Performance - 7.5 ns tPD, 95 MHz External Operation
  - Enhanced Logic Flexibility
  - Backward Compatible with ATV750/L Software and Hardware
- New Flip-Flop Features
  - D- or T-Type
  - Product Term or Direct Input Pin Clocking
- Highest Density Programmable Logic Available in a 24-Pin Package
- Increased Logic Flexibility
  - 42 Array Inputs, 20 Sum Terms and 20 Flip-Flops
- Enhanced Output Logic Flexibility
  - All 20 Flip-Flops Feed Back Internally
  - 10 Flip-Flops are Also Available as Outputs
- Low Power ATV750BL - 1.0 mA Standby (Typical)
- Reprogrammable - 100% Tested for Programming
- Full Military, Commercial and Industrial Temperature Ranges
- 24-Pin, 0.300" DIP, 24-Lead SOIC, and 28-Lead Surface Mount Packages

Logic Diagram



Description

The ATV750Bs are twice as powerful as most other 24-pin programmable logic devices. Increased product terms, sum terms, flip-flops and output logic configurations translate into more usable gates. High speed logic and uniform, predictable delays guarantee fast in-system performance.

Each of the ATV750B's 22 logic pins can be used as an input. Ten of these can be used as inputs, outputs or bi-directional I/O pins. Each flip-flop is individually configurable as either D- or T-type. Each flip-flop output is fed back into the array independently. This allows burying of all the sum terms and flip-flops.

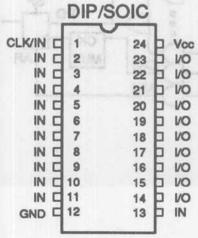
There are 171 total product terms available. A variable format is used to assign between four to eight product terms per sum term. There are two sum terms per output, providing added

(continued on next page)

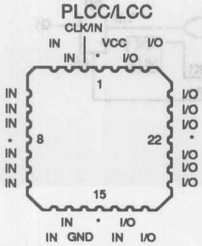
Pin Configurations

Pin Name	Function
CLK	Clock
IN	Logic Inputs
I/O	Bidirectional Buffers
*	No Internal Connection
Vcc	+5 V Supply

DIP/SOIC



PLCC/LCC



Description (Continued)

flexibility. Much more logic can be replaced by this device than by any other 24-pin PLD. With 20 sum terms and flip-flops, complex state machines are easily implemented with logic to spare.

Product terms provide individual clocks and asynchronous resets for each flip-flop. Each flip-flop may also be individually configured to have direct input pin controlled clocking. Each output has its own enable product term. One product term provides a common synchronous preset for all flip-flops. Register preload functions are provided to simplify testing. All registers automatically reset upon power up.

The ATV750BL is a low power device with speeds as fast as 10 ns. The ATV750BL provides the optimum low power PLD solution, with full CMOS output levels. This device significantly reduces total system power, thereby allowing battery-powered operation.

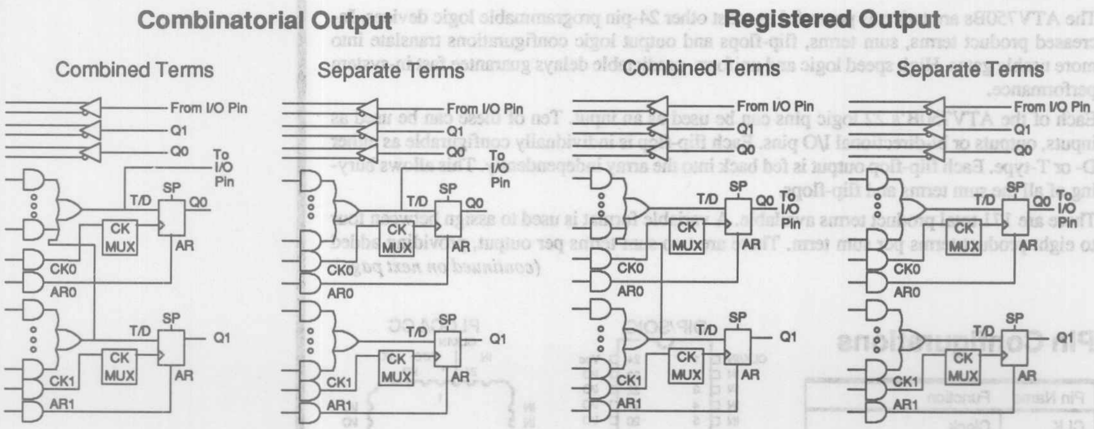
Absolute Maximum Ratings*

Temperature Under Bias	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-2.0 V to +7.0 V(1)
Voltage on Input Pins with Respect to Ground During Programming	-2.0 V to +14.0 V(1)
Programming Voltage with Respect to Ground	-2.0 V to +14.0 V(1)
Integrated UV Erase Dose	7258 W·sec/cm²

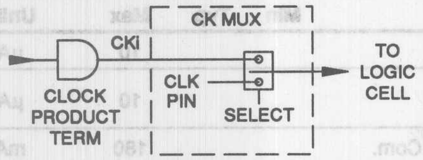
*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Note:
1. Minimum voltage is -0.6 V dc which may undershoot to -2.0 V for pulses of less than 20 ns. Maximum output pin voltage is V_{CC}+0.75 V dc which may overshoot to +7.0 V for pulses of less than 20 ns.

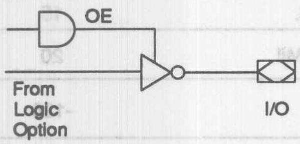
Logic Options

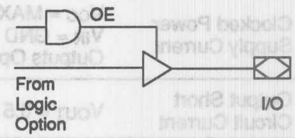


1



Output Options





D.C. and A.C. Operating Conditions (1)

	Commercial -7, -10, -15	Commercial -25	Industrial	Military
Operating Temperature (Case)	0°C - 70°C	0°C - 70°C	-40°C - 85°C	-55°C - 125°C
V_{CC} Power Supply	5 V ± 5%	5 V ± 10%	5 V ± 10%	5 V ± 10%

Note: 1. See ordering information for valid speed and temperature combination.

Operating Modes

Mode	24-Pin DIP/SOIC	1	5	8	11	13	I/Os	V_{CC} (24)
	28-Lead SMD	2	6	10	13	16	I/Os	V_{CC} (28)
"PLD"		X (1)	X	X	X	X	I/O	4.5 - 5.5 V
Program		V_{PP}	X/V_H (2)	X	X/V_H	V_{PP}	D_{IN}	6 V
PGM Verify		V_{PP}	X/V_H	X	X/V_H	V_{IL}	D_{OUT}	5 V
PGM Inhibit		V_{PP}	X/V_H	X	X/V_H	V_{IH}	High Z	5-6 V
Preload #1		X	X	V_H	X	V_{IL}	D_{IN}	4.5 - 5.5 V
Preload #2		X	X	V_H	X	V_{IH}	D_{IN}	4.5 - 5.5 V

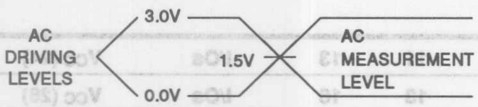
Notes: 1. X can be V_{IL} or V_{IH}.
2. V_H = 10.25 V to 10.75 V.

D.C. Characteristics

Symbol	Parameter	Condition			Min	Typ	Max	Units
I_{LI}	Input Load Current	V_{IN} = -0.1 V to V_{CC}+1 V					10	µA
I_{LO}	Output Leakage Current	V_{OUT} = -0.1 V to V_{CC}+0.1 V					10	µA
I_{CC}	Power Supply Current, Standby	VCC = MAX, V_{IN} = GND or Vcc, f = 0 MHz, Outputs Open	ATV750B	Com.			180	mA
				Ind.,Mil.			190	mA
			ATV750BL	Com.		1	4	mA
				Ind.,Mil.		1	5	mA
I_{CC2}	Clocked Power Supply Current	V_{CC} = MAX, V_{IN} = GND or V_{CC}, Outputs Open	ATV750BL	Com.			15	mA/MHz (2)
				Ind.,Mil.			20	mA/MHz (2)
I_{OS} (1)	Output Short Circuit Current	V_{OUT} = 0.5 V					-120	mA
V_{IL}	Input Low Voltage	4.5 ≤ V_{CC} ≤ 5.5 V			-0.6		0.8	V
V_{IH}	Input High Voltage				2.0		V_{CC}+0.75	V
V_{OL}	Output Low Voltage	VIN = V_{IH} or V_{IL}, V_{CC} = MIN	I_{OL} = 16 mA Com.,Ind.				0.5	V
			I_{OL} = 12 mA Mil.				0.5	V
			I_{OL} = 24 mA Com.				0.8	V
V_{OH}	Output High Voltage	VIN = V_{IH} or V_{IL}, V_{CC} = MIN	I_{OH} = -100 µA		V_{CC}-0.3			V
			I_{OH} = -4.0 mA		2.4			V

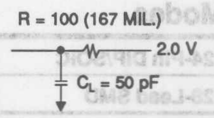
Notes: 1. Not more than one output at a time should be shorted. Duration of short circuit test should not exceed 30 sec.
2. See I_{CC} versus frequency characterization curves.

Input Test Waveforms and Measurement Levels

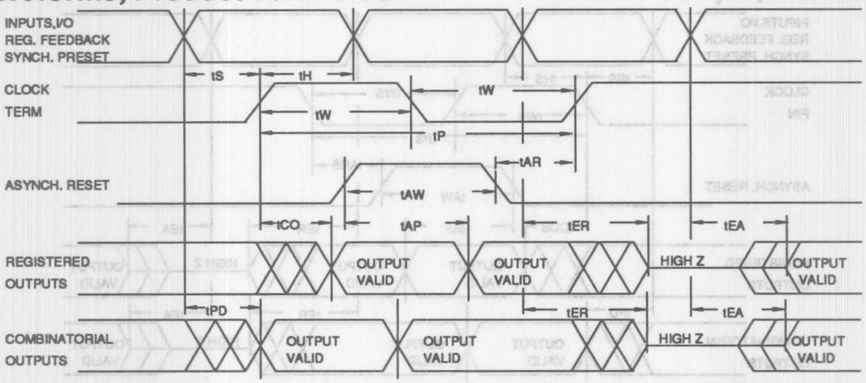


tR, tF < 3 ns (10% to 90%)

Output Test Load



A.C. Waveforms, Product Term Clock[1]



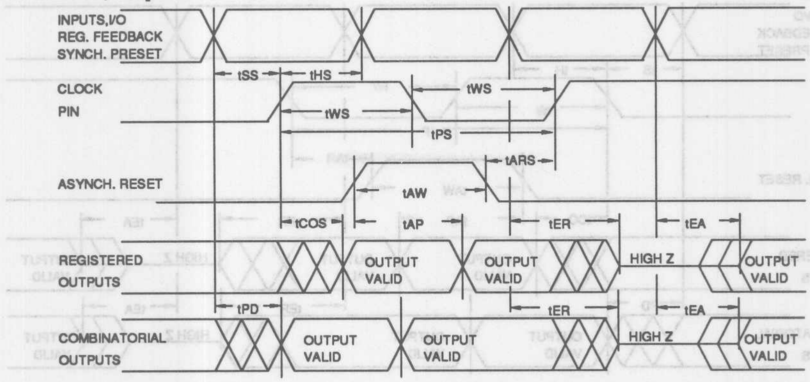
Note: 1. Timing measurement reference is 1.5 V. Input AC driving levels are 0.0 V and 3.0 V, unless otherwise specified.

A.C. Characteristics, Product Term Clock[1]

Symbol	Parameter	-7 Min	-7 Max	B/BL -10 Min	B/BL -10 Max	-15 Min	-15 Max	-25 Min	-25 Max	Units
t_{PD}	Input or Feedback to Non-Registered Output		7.5		10		15		25	ns
t_{EA}	Input to Output Enable		7.5		10		15		25	ns
t_{ER}	Input to Output Disable		7.5		10		15		25	ns
t_{CO}	Clock to Output	3	7.5	4	10	5	15	6	22	ns
t_{CF}	Clock to Feedback	1	5	4	7.5	5	9	5	10	ns
t_S	Input Setup Time	3		4/5.5		7		12		ns
t_{SF}	Feedback Setup Time	3		4/5.5		7		7		ns
t_H	Hold Time	1		2/4		5		5		ns
t_P	Clock Period	7		11		14		17		ns
t_W	Clock Width	3.5		5.5		7		8.5		ns
F_{MAX}	External Feedback $1/(t_S+t_{CO})$		95		71/64		45		29	MHz
	Internal Feedback $1/(t_{SF}+t_{CF})$		125		86/76		62		58	MHz
	No Feedback $1/(t_P)$		142		90		71		58	MHz
t_{AW}	Asynchronous Reset Width	5		10		15		20		ns
t_{AR}	Asynchronous Reset Recovery Time	3		10		15		20		ns
t_{AP}	Asynchronous Reset to Registered Output Reset		8		12		15		25	ns
t_{SP}	Setup Time, Synchronous Preset	4		7		8		15		ns

Note: 1. See ordering information for valid part numbers.

A.C. Waveforms, Input Pin Clock[1]

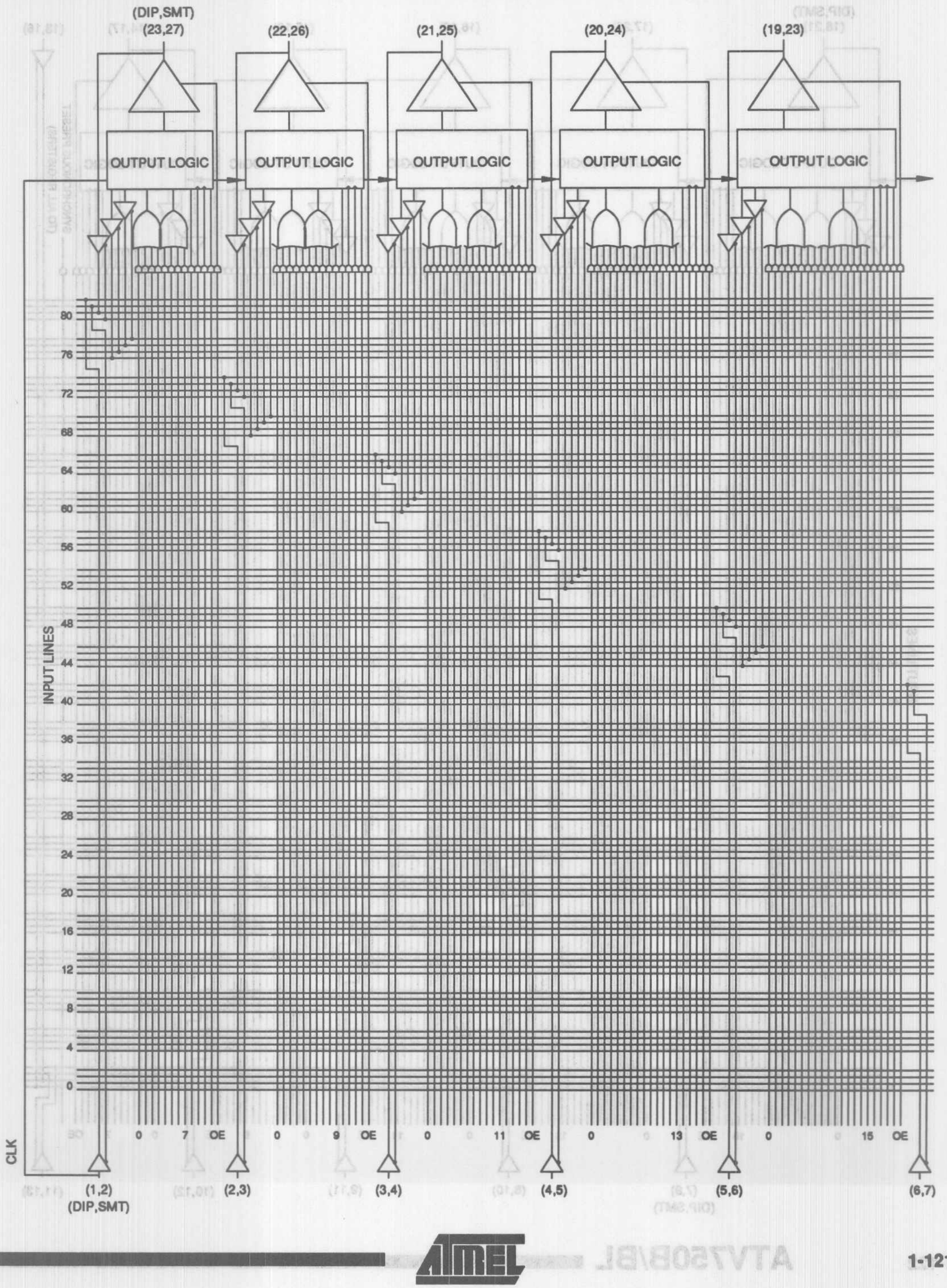


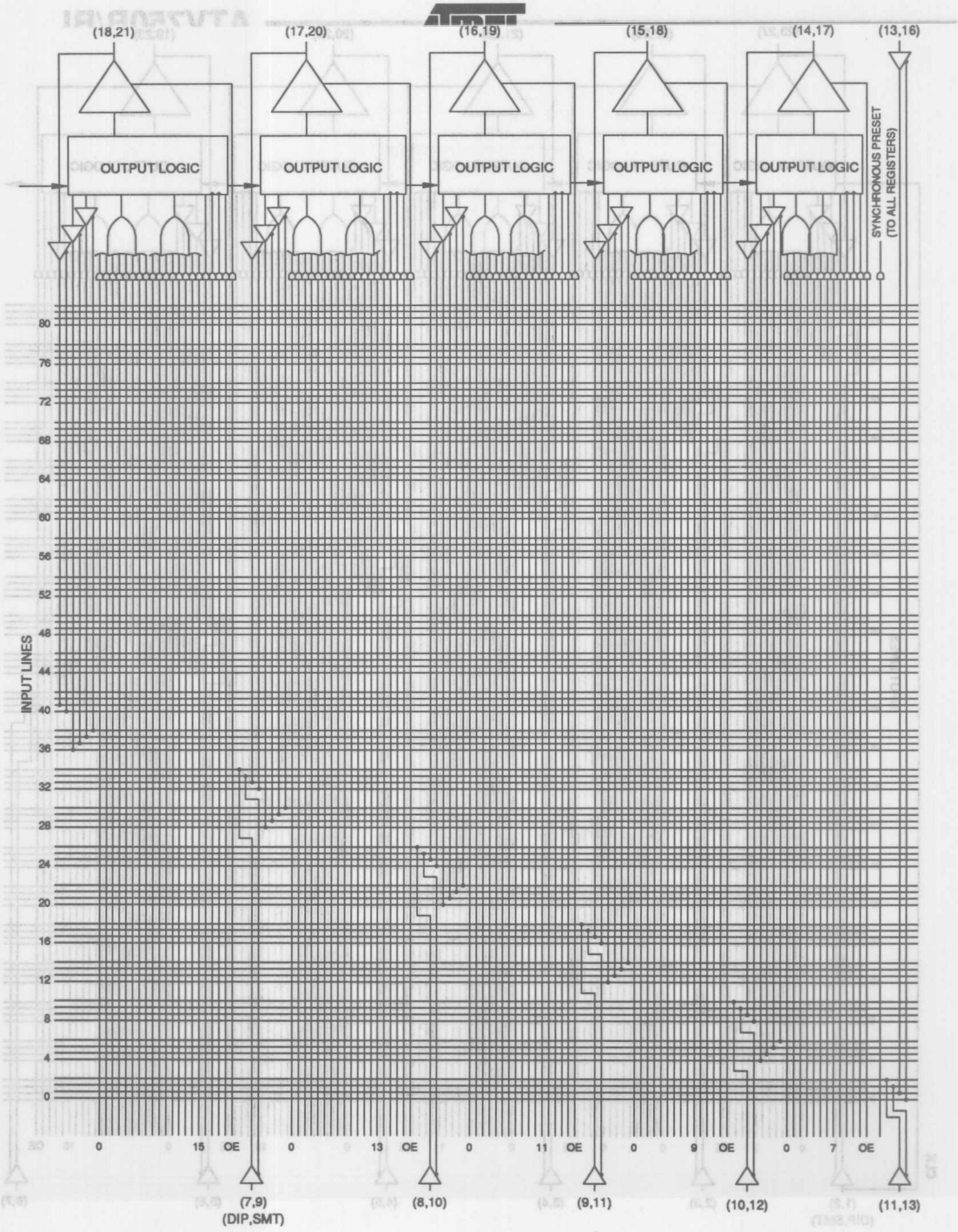
Note: 1. Timing measurement reference is 1.5 V. Input AC driving levels are 0.0 V and 3.0 V, unless otherwise specified.

A.C. Characteristics, Input Pin Clock

Symbol	Parameter	-7		B/BL -10		-15		-25		Units
		Min	Max	Min	Max	Min	Max	Min	Max	
t_{PD}	Input or Feedback to Non-Registered Output		7.5		10		15		25	ns
t_{EA}	Input to Output Enable		7.5		10		15		25	ns
t_{ER}	Input to Output Disable		7.5		10		15		25	ns
t_{COS}	Clock to Output	0	6.5	0	7	0	10	0	15	ns
t_{CFS}	Clock to Feedback	0	3.5	0	5	0	5.5	0	7	ns
t_{SS}	Input Setup Time	4		5/6.5		7		9		ns
t_{SFS}	Feedback Setup Time	4		5/6.5		7		9		ns
t_{HS}	Hold Time	0		0		0		0		ns
t_{PS}	Clock Period	7		10		12		16		ns
t_{WS}	Clock Width	3.5		5		6		8		ns
F_{MAXS}	External Feedback $1/(t_{SS}+t_{COS})$		95		83/74		58		41	MHz
	Internal Feedback $1/(t_{SFS}+t_{CFS})$		133		100/86		80		62	MHz
	No Feedback $1/(t_{PS})$		142		100		83		62	MHz
t_{AW}	Asynchronous Reset Width	5		10		15		20		ns
t_{ARS}	Asynchronous Reset Recovery Time	5		10		15		25		ns
t_{AP}	Asynchronous Reset to Registered Output Reset		8		10		15		25	ns
t_{SPS}	Setup Time, Synchronous Preset	5		5/9		11		15		ns

Functional Logic Diagram ATV750B, Upper Half



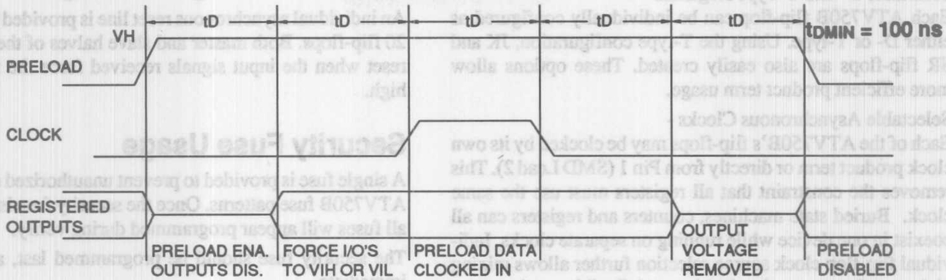


(18,21)
(17,20)
(16,19)
(15,18)
(14,17)
(13,16)
OUTPUT LOGIC
OUTPUT LOGIC
OUTPUT LOGIC
OUTPUT LOGIC
OUTPUT LOGIC
SYNCHRONOUS PRESET
(TO ALL REGISTERS)
INPUT LINES
80
76
72
68
64
60
56
52
48
44
40
36
32
28
24
20
16
12
8
4
0
0
15
OE
0
13
OE
0
11
OE
0
9
OE
0
7
OE
(7,9)
(DIP,SMT)
(8,10)
(9,11)
(10,12)
(11,13)

1

Preload of Registered Outputs

The ATV750B's registers are provided with circuitry to allow loading of each register asynchronously with either a high or a low. This feature will simplify testing since any state can be forced into the registers to control test sequencing. A V_{IH} level on the I/O pin will force the register high; a V_{IL} will force it low, independent of the output polarity. The PRELOAD state is entered by placing a 10.25 V to 10.75 V signal on pin 8 on DIPs, and lead 10 on SMDs. When the clock term is pulsed high, the data on the I/O pins is placed into the register chosen by the Select Pin.



Level forced on registered output pin during PRELOAD cycle	Select Pin State	Register #0 state after cycle	Register #1 State after cycle
V_{IH}	Low	High	X
V_{IL}	Low	Low	X
V_{IH}	High	X	High
V_{IL}	High	X	Low

Power Up Reset

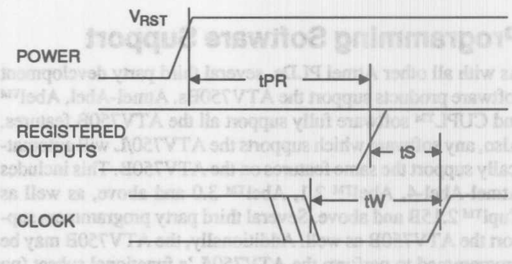
The registers in the ATV750Bs are designed to reset during power up. At a point delayed slightly from V_{CC} crossing V_{RST}, all registers will be reset to the low state. The output state will depend on the polarity of the output buffer.

This feature is critical for state machine initialization. However, due to the asynchronous nature of reset and the uncertainty of how V_{CC} actually rises in the system, the following conditions are required:

1) The V_{CC} rise must be monotonic,

2) After reset occurs, all input and feedback setup times must be met before driving the clock term high, and

3) The signals from which the clock is derived must remain stable during t_{PR}.



Parameter	Description	Typ	Max	Units
t_{PR}	Power-Up Reset Time	600	1000	ns
V_{RST}	Power-Up Reset Voltage	3.8	4.5	V

Pin Capacitance (f = 1 MHz, T = 25°C) [1]

	Typ	Max	Units	Conditions
C_{IN}	5	8	pF	V_{IN} = 0 V
C_{OUT}	6	8	pF	V_{OUT} = 0 V

Note: 1. Typical values for nominal supply voltage. This parameter is only sampled and is not 100% tested.

Using the ATV750B's Many Advanced Features

The ATV750B's advanced flexibility packs more usable gates into 24 pins than any other logic device. The ATV750Bs start with the popular 22V10 architecture, and add several enhanced features:

- Selectable D- and T-Type Registers -
  Each ATV750B flip-flop can be individually configured as either D- or T-type. Using the T-type configuration, JK and SR flip-flops are also easily created. These options allow more efficient product term usage.
- Selectable Asynchronous Clocks -
  Each of the ATV750B's flip-flops may be clocked by its own clock product term or directly from Pin 1 (SMD Lead 2). This removes the constraint that all registers must use the same clock. Buried state machines, counters and registers can all coexist in one device while running on separate clocks. Individual flip-flop clock source selection further allows mixing higher performance pin clocking and flexible product term clocking within one design.
- A Full Bank of Ten More Registers -
  The ATV750B provides two flip-flops per output logic cell for a total of 20. Each register has its own sum term, its own reset term and its own clock term.
- Independent I/O Pin and Feedback Paths -
  Each I/O pin on the ATV750B has a dedicated input path. Each of the 20 registers has its own feedback terms into the array as well. This feature, combined with individual product terms for each I/O's output enable, facilitates true bi-directional I/O design.

Programming Software Support

As with all other Atmel PLDs, several third party development software products support the ATV750Bs. Atmel-Abel, Abel™ and CUPL™ software fully support all the ATV750B features. Also, any software which supports the ATV750/L will automatically support the same features on the ATV750B. This includes Atmel-Abel-4, Abel™ 2.1, Abel™ 3.0 and above, as well as Cupl™ 2.15B and above. Several third party programmers support the ATV750B as well. Additionally, the ATV750B may be programmed to perform the ATV750/L's functional subset (no T-type flip-flops or pin clocking) using the ATV750/L JEDEC file. In this case, the ATV750B becomes a direct replacement or speed upgrade for the ATV750/L. The ATV750/L programming algorithm is different from the ATV750B/BL algorithm. Choose the appropriate device in your programmer menu to ensure proper programming.

Synchronous Preset and Asynchronous Reset

One synchronous preset line is provided for all 20 registers in the ATV750B. The appropriate input signals to cause the internal clocks to go to a high state must be received during a synchronous preset. Appropriate setup and hold times must be met, as shown in the switching waveform diagram.

An individual asynchronous reset line is provided for each of the 20 flip-flops. Both master and slave halves of the flip-flops are reset when the input signals received force the internal resets high.

Security Fuse Usage

A single fuse is provided to prevent unauthorized copying of the ATV750B fuse patterns. Once the security fuse is programmed, all fuses will appear programmed during verify.

The security fuse should be programmed last, as its effect is immediate.

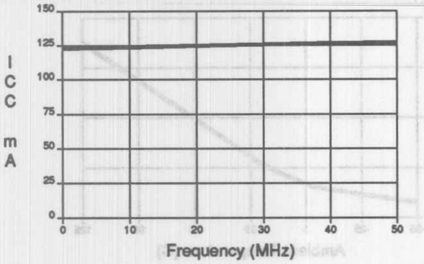
Erasure Characteristics

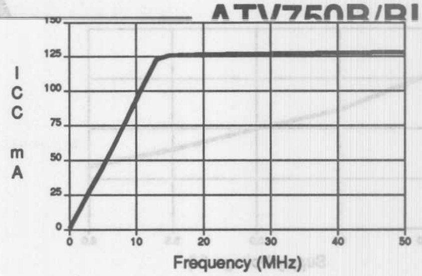
The entire memory array of an ATV750B is erased after exposure to ultraviolet light at a wavelength of 2537 Å. Complete erasure is assured after a minimum of 20 minutes exposure using 12,000 $\mu W/cm^2$ intensity lamps spaced one inch away from the chip. Minimum erase time for lamps at other intensity ratings can be calculated from the minimum integrated erasure dose of 15 $W{\cdot}sec/cm^2$. To prevent unintentional erasure, an opaque label is recommended to cover the clear window on any UV erasable PLD which will be subjected to continuous fluorescent indoor lighting or sunlight.

Atmel CMOS PLDs

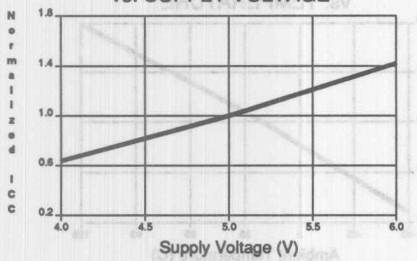
The ATV750B utilizes an advanced 0.65-micron CMOS EPROM technology. This technology's state of the art features are the optimum combination for PLDs:

- CMOS technology provides high speed, low power, and high noise immunity.
- EPROM technology is the most cost effective method for producing PLDs - surpassing bipolar fusible link technology in low cost, while providing the necessary reprogrammability.
- EPROM reprogrammability, which is 100% tested before shipment, provides inherently better programmability and reliability than one-time fusible PLDs.

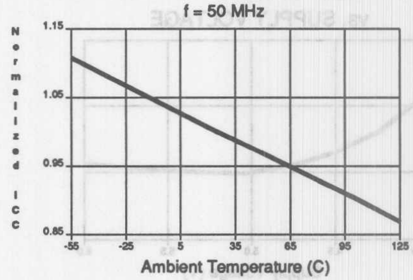




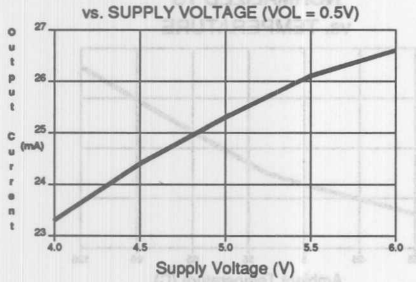
NORMALIZED SUPPLY CURRENT
vs. SUPPLY VOLTAGE



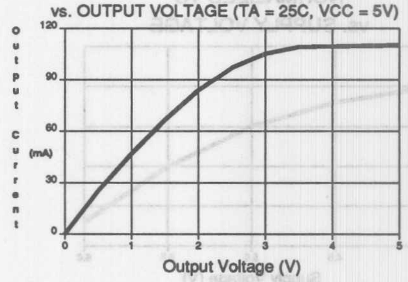
NORMALIZED ICC vs. AMBIENT TEMP.
f = 50 MHz



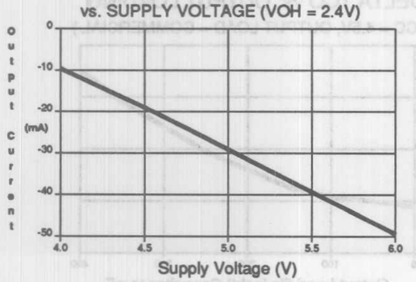
OUTPUT SINK CURRENT
vs. SUPPLY VOLTAGE (VOL = 0.5V)



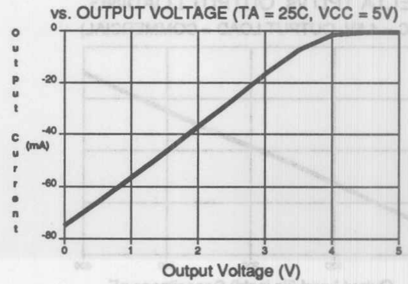
OUTPUT SINK CURRENT
vs. OUTPUT VOLTAGE (TA = 25C, VCC = 5V)



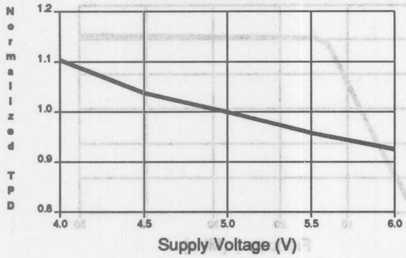
OUTPUT SOURCE CURRENT
vs. SUPPLY VOLTAGE (VOH = 2.4V)



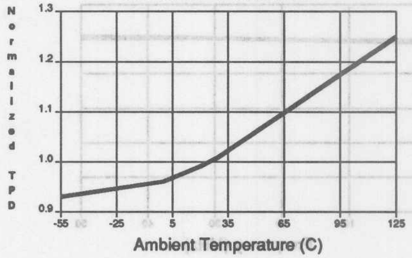
OUTPUT SOURCE CURRENT
vs. OUTPUT VOLTAGE (TA = 25C, VCC = 5V)



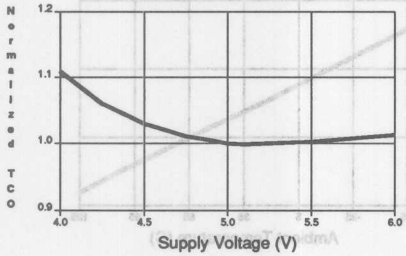
NORMALIZED TPD vs. SUPPLY VOLTAGE



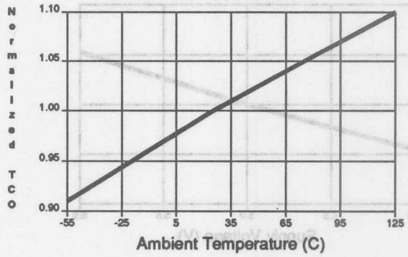
NORMALIZED TPD vs. TEMPERATURE



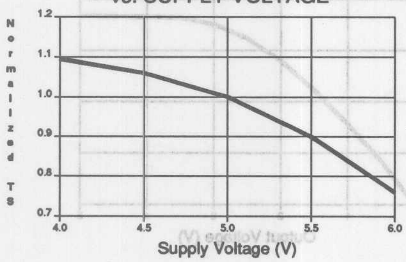
NORMALIZED TCO vs. SUPPLY VOLTAGE



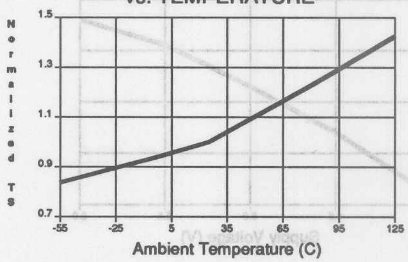
NORMALIZED TCO vs. TEMPERATURE



NORMALIZED TS vs. SUPPLY VOLTAGE

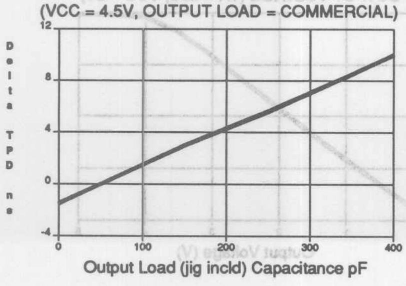


NORMALIZED TS vs. TEMPERATURE



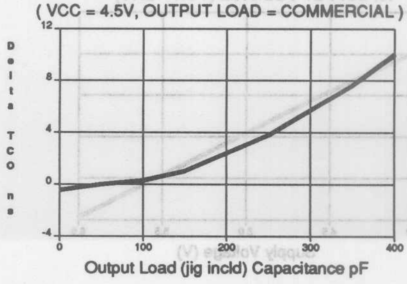
DELTA TPD vs. OUTPUT LOADING

(VCC = 4.5V, OUTPUT LOAD = COMMERCIAL)



DELTA TCO vs. OUTPUT LOADING

(VCC = 4.5V, OUTPUT LOAD = COMMERCIAL)



Ordering Information

t_{PD} (ns)	t_{COS} (ns)	Ext. f_{MAXS} (MHz)	Ordering Code	Package	Operation Range
7.5	6.5	95	ATV750B-7DC ATV750B-7JC ATV750B-7PC	24DW3 28J 24P3	Commercial (0°C to 70°C)
10	7	83	ATV750B-10DC ATV750B-10JC ATV750B-10KC ATV750B-10PC ATV750B-10SC	24DW3 28J 28KW 24P3 24S	Commercial (0°C to 70°C)
			ATV750B-10DI ATV750B-10JI ATV750B-10KI ATV750B-10PI ATV750B-10SI	24DW3 28J 28KW 24P3 24S	Industrial (-40°C to 85°C)
			ATV750B-10DM ATV750B-10KM ATV750B-10LM	24DW3 28KW 28LW	Military (-55°C to 125°C)
			ATV750B-10DM/883 ATV750B-10KM/883 ATV750B-10LM/883	24DW3 28KW 28LW	Military/883C (-55°C to 125°C) Class B, Fully Compliant
15	10	58	ATV750B-15DC ATV750B-15JC ATV750B-15KC ATV750B-15PC ATV750B-15SC	24DW3 28J 28KW 24P3 24S	Commercial (0°C to 70°C)
			ATV750B-15DI ATV750B-15JI ATV750B-15KI ATV750B-15PI ATV750B-15SI	24DW3 28J 28KW 24P3 24S	Industrial (-40°C to 85°C)
			ATV750B-15DM ATV750B-15KM ATV750B-15LM	24DW3 28KW 28LW	Military (-55°C to 125°C)
			ATV750B-15DM/883 ATV750B-15KM/883 ATV750B-15LM/883	24DW3 28KW 28LW	Military/883C (-55°C to 125°C) Class B, Fully Compliant
25	15	41	ATV750B-25DC ATV750B-25JC ATV750B-25KC ATV750B-25PC ATV750B-25SC	24DW3 28J 28KW 24P3 24S	Commercial (0°C to 70°C)
			ATV750B-25DI ATV750B-25JI ATV750B-25KI ATV750B-25PI ATV750B-25SI	24DW3 28J 28KW 24P3 24S	Industrial (-40°C to 85°C)

Ordering Information

t_{PD} (ns)	t_{COS} (ns)	Ext. f_{MAXS} (MHz)	Ordering Code	Package	Operation Range
10	7	74	ATV750BL-10DC	24DW3	Commercial (0°C to 70°C)
			ATV750BL-10JC	28J	
			ATV750BL-10PC	24P3	
15	10	58	ATV750BL-15DC	24DW3	Commercial (0°C to 70°C)
			ATV750BL-15JC	28J	
			ATV750BL-15KC	28KW	
			ATV750BL-15PC	24P3	
			ATV750BL-15SC	24S	
			ATV750BL-15DI	24DW3	Industrial (-40°C to 85°C)
			ATV750BL-15JI	28J	
			ATV750BL-15KI	28KW	
			ATV750BL-15PI	24P3	
			ATV750BL-15SI	24S	
			ATV750BL-15DM	24DW3	Military (-55°C to 125°C)
			ATV750BL-15KM	28KW	
			ATV750BL-15LM	28LW	
			ATV750BL-15DM/883	24DW3	Military/883C (-55°C to 125°C) Class B, Fully Compliant
			ATV750BL-15KM/883	28KW	
			ATV750BL-15LM/883	28LW	
25	15	41	ATV750BL-25DC	24DW3	Commercial (0°C to 70°C)
			ATV750BL-25JC	28J	
			ATV750BL-25KC	28KW	
			ATV750BL-25PC	24P3	
			ATV750BL-25SC	24S	
			ATV750BL-25DI	24DW3	Industrial (-40°C to 85°C)
			ATV750BL-25JI	28J	
			ATV750BL-25KI	28KW	
			ATV750BL-25PI	24P3	
			ATV750BL-25SI	24S	

Package Type	
24DW3	24 Lead, 0.300" Wide, Windowed, Ceramic Dual Inline Package (Cerdip)
28J	28 Lead, Plastic J-Leaded Chip Carrier OTP (PLCC)
28KW	28 Lead, Windowed, Ceramic J-Leaded Chip Carrier (JLCC)
28LW	28 Pad, Windowed, Ceramic Leadless Chip Carrier (LCC)
24P3	24 Lead, 0.300" Wide, Plastic Dual Inline Package OTP (PDIP)
24S	24 Lead, 0.300" Wide, Plastic Gull Wing Small Outline OTP (SOIC)

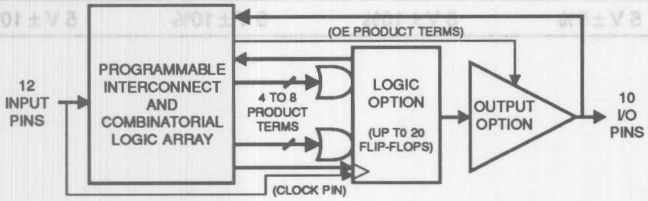
High Speed UV Erasable Programmable Logic Device

Advanced Information

Features

- Quarter Power Equivalent of ATV750B/BL - 90 mA Maximum
- Low Power ATV750BQL - 1.0 mA Standby (Typical)
- Advanced, High Speed Programmable Logic Device
  - 15 ns Maximum Pin-To-Pin Delay
  - Enhanced Logic Flexibility
  - Backward Compatible with ATV750/L Software and Hardware
- New Flip-Flop Features
  - D- or T-Type
  - Product Term or Direct Input Pin Clocking
- Highest Density Programmable Logic Available in a 24-Pin Package
- Increased Logic Flexibility
  - 42 Array Inputs, 20 Sum Terms and 20 Flip-Flops
- Enhanced Output Logic Flexibility
  - All 20 Flip-Flops Feed Back Internally
  - 10 Flip-Flops are Also Available as Outputs
- Reprogrammable - 100% Tested for Programming
- Full Military, Commercial and Industrial Temperature Ranges
- 24-Pin, 0.300" DIP, 24-Lead SOIC, and 28-Lead Surface Mount Packages

Logic Diagram



Description

The ATV750BQs are twice as powerful at lower current requirements than most other 24-pin programmable logic devices. Increased product terms, sum terms, flip-flops and output logic configurations translate into more usable gates. High speed logic and uniform, predictable delays guarantee fast in-system performance.

Each of the ATV750BQ's 22 logic pins can be used as an input. Ten of these can be used as inputs, outputs or bi-directional I/O pins. Each flip-flop is individually configurable as either D- or T-type. Each flip-flop output is fed back into the array independently. This allows burying of all the sum terms and flip-flops.

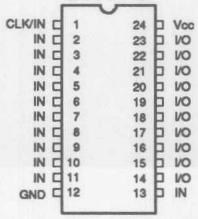
There are 171 total product terms available. A variable format is used to assign between four to eight product terms per sum term. There are two sum terms per output, providing added

(continued on next page)

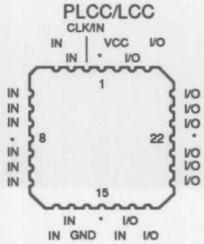
Pin Configurations

Pin Name	Function
CLK	Clock
IN	Logic Inputs
I/O	Bidirectional Buffers
*	No Internal Connection
Vcc	+5 V Supply

DIP/SOIC



PLCC/LCC



Description (Continued)

flexibility. Much more logic can be replaced by this device than by any other 24-pin PLD. With 20 sum terms and flip-flops, complex state machines are easily implemented with logic to spare.

Product terms provide individual clocks and asynchronous resets for each flip-flop. Each flip-flop may also be individually configured to have direct input pin controlled clocking. Each output has its own enable product term. One product term provides a common synchronous preset for all flip-flops. Register preload functions are provided to simplify testing. All registers automatically reset upon power up.

The ATV750BQL is a low power device with speeds as fast as 15 ns. The ATV750BQL provides the optimum low power PLD solution, with full CMOS output levels. Typical standby current is only 1 mA. This device significantly reduces total system power, thereby allowing battery-powered operation.

D.C. and A.C. Operating Conditions

	Commercial -15	Commercial -25	Industrial -25	Military -25
Operating Temperature (Case)	0°C - 70°C	0°C - 70°C	-40°C - 85°C	-55°C - 125°C
V_{CC} Power Supply	5 V ± 5%	5 V ± 10%	5 V ± 10%	5 V ± 10%

1

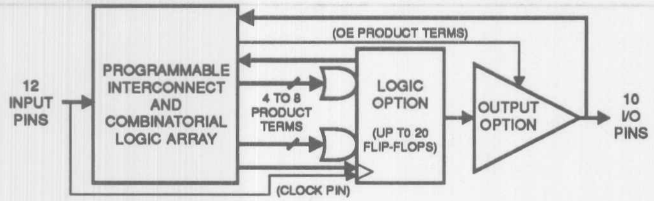
High Speed UV Erasable Programmable Logic Device

Advanced Information

Features

- 3-Volt Operation - Low System Power Requirement
- Wide Vcc Range
  - Vcc = 3.0 V to 5.25 V (Commercial)
  - Vcc = 3.0 V to 5.5 V (Industrial)
- Advanced, High Speed Programmable Logic Device
  - 10 ns Maximum Pin-To-Pin Delay
  - Enhanced Logic Flexibility
  - Architecture Identical to ATV750B/BL
  - Backward Compatible with ATV750/L Software and Hardware
- Low Power, Low Voltage ATLV750BL - 0.5 mA Standby (Typical) at 3.6 V
- New Flip-Flop Features
  - D- or T-Type
  - Product Term or Direct Input Pin Clocking
- Highest Density Programmable Logic Available in a 24-Pin Package
- Increased Logic Flexibility
  - 42 Array Inputs, 20 Sum Terms and 20 Flip-Flops
- Enhanced Output Logic Flexibility
  - All 20 Flip-Flops Feed Back Internally
  - 10 Flip-Flops are Also Available as Outputs
- Reprogrammable - 100% Tested for Programming
- Full Commercial and Industrial Temperature Ranges
- 24-Pin, 0.300" DIP, 24-Lead SOIC, and 28-Lead Surface Mount Packages

Logic Diagram



Description

The ATLV750Bs are twice as powerful as most other 24-pin programmable logic devices. Increased product terms, sum terms, flip-flops and output logic configurations translate into more usable gates. High speed logic and uniform, predictable delays guarantee fast in-system performance.

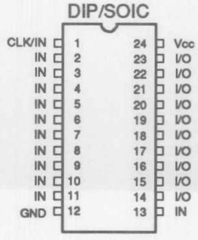
Each of the ATLV750B's 22 logic pins can be used as an input. Ten of these can be used as inputs, outputs or bi-directional I/O pins. Each flip-flop is individually configurable as either

(continued on next page)

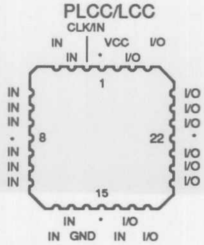
Pin Configurations

Pin Name	Function
CLK	Clock
IN	Logic Inputs
I/O	Bidirectional Buffers
*	No Internal Connection
Vcc	+5 V Supply

DIP/SOIC



PLCC/LCC



Description (Continued)

D- or T-type. Each flip-flop output is fed back into the array independently. This allows burying of all the sum terms and flip-flops.

There are 171 total product terms available. A variable format is used to assign between four to eight product terms per sum term. There are two sum terms per output, providing added flexibility. Much more logic can be replaced by this device than by any other 24-pin PLD. With 20 sum terms and flip-flops, complex state machines are easily implemented with logic to spare.

Product terms provide individual clocks and asynchronous resets for each flip-flop. Each flip-flop may also be individually configured to have direct input pin controlled clocking. Each output has its own enable product term. One product term provides a common synchronous preset for all flip-flops. Register preload functions are provided to simplify testing. All registers automatically reset upon power up.

The ATLV750B and ATLV750BL are low voltage and low power devices with speeds as fast as 10 ns. Architecturally identical to the ATV750B/BL, the ATLV750B/BL satisfies most low voltage, low power portable design requirements. The ATLV750BL provides the optimum low power PLD solution, with full CMOS output levels. Standby power dissipation is as low as 1.8 mW at 3.6-V operation. This device significantly reduces total system power, thereby allowing battery-powered operation.

D.C. and A.C. Operating Conditions

	Commercial -10, -15	Industrial -10, -15
Operating Temperature (Case)	$0^{o}C$ - $70^{o}C$	$-40^{o}C$ - $85^{o}C$
V_{CC} Power Supply	3.0 V to 5.25 V	3.0 V to 5.5 V

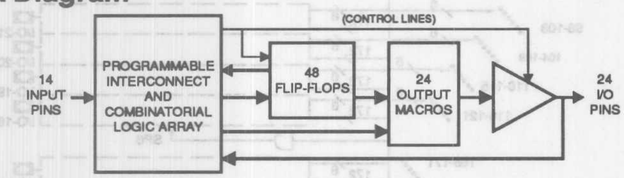
1

High Density UV Erasable Programmable Logic Device

Features

- **Third Generation Programmable Logic Structure**
  Easily Achieves Gate Utilization Factors of 80 Percent
- **Increased Logic Flexibility**
  86 Inputs and 72 Sum Terms
- **Flexible Output Macrocell**
  48 Flip-Flops - 2 per Macrocell
  3 Sum Terms - Can Be OR'ed and Shared
- **High Speed**
- **Low Power - Less than 0.5 mA Typical (ATV2500L)**
- **Multiple Feedback Paths Provide for Buried State Machines and I/O Bus Compatibility**
- **Asynchronous Clocks and Resets**
  Multiple Synchronous Presets - One per Four or Eight Flip-Flops
- **Proven and Reliable High Speed CMOS EPROM Process**
  2000 V ESD Protection
  200 mA Latchup Immunity
- **Reprogrammable - Tested 100% for Programmability**
- **40-pin Dual-in-line and 44-Lead Surface Mount Packages**

Block Diagram



Description

The ATV2500H/L is the most powerful programmable logic device available in a 40-pin package. Increased product terms, sum terms, and flip-flops translate into many more usable gates. High gate utilization is easily obtainable.

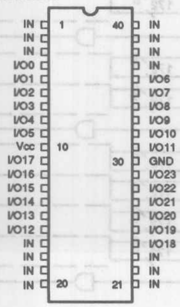
The ATV2500H/L is organized around a global bus. All pin and feedback terms are always available to every logic cell. Each of the 38 logic pins and their complements are array inputs, as well as the true and false outputs of each of the 48 flip-flops.

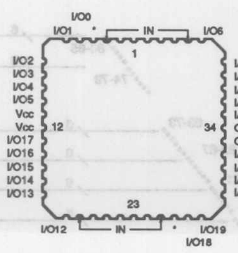
There are 416 product terms available. Four product terms are input to each sum term. The three sum terms per logic cell can be combined to provide up to 12 product terms, combinatorial and registered. Independent of output configuration, the two flip-flops are always usable, and always have at least four product term inputs.

Product terms are available providing asynchronous resets, flip-flop clocks, and output enables. One reset and one clock term are provided per flip-flop, with one enable term per output. Eight product terms provide local synchronous presets, divided up into banks of four and eight flip-flops. Register preload and buried register observability simplify testing. The device has an internal power up clear function.

Pin Configurations

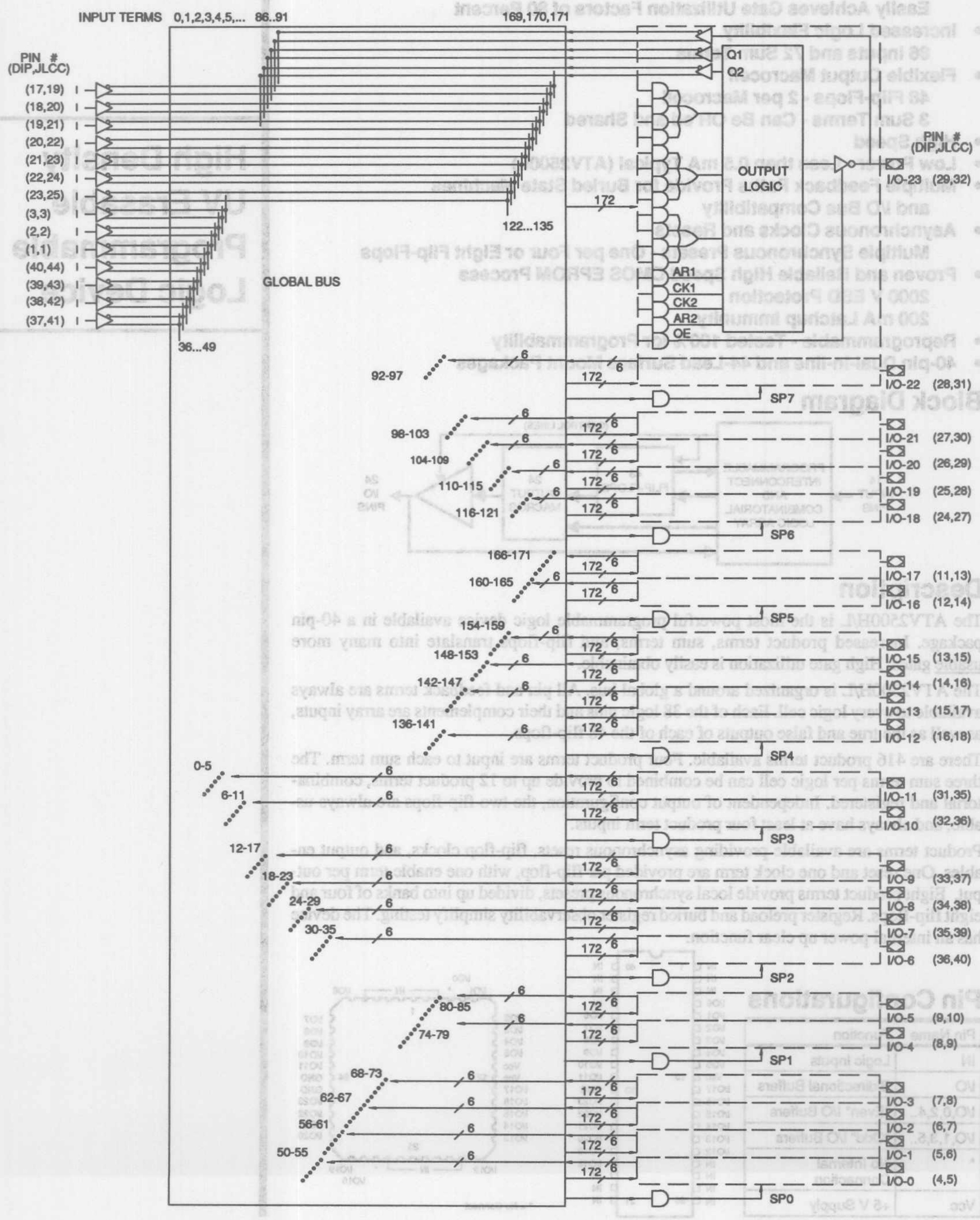
Pin Name	Function
IN	Logic Inputs
I/O	Bidirectional Buffers
I/O,0,2,4..	"Even" I/O Buffers
I/O,1,3,5..	"Odd" I/O Buffers
*	No Internal Connection
Vcc	+5 V Supply





* = No Connect

Functional Logic Diagram ATV2500H/L



Functional Logic Diagram Description

The ATV2500H/L Functional Logic Diagram describes the interconnections between the input, feedback pins and logic cells. All interconnections are routed through the global bus.

The ATV2500H/L is a straightforward and uniform PLD. The 24 macrocells are numbered 0 through 23. Each macrocell contains 17 AND gates. All AND gates have 172 inputs. The five lower product terms provide AR1, CK1, CK2, AR2, and OE. These are: one asynchronous reset and clock per flip-flop, and an output enable. The top 12 product terms are grouped into three sum terms, which are used as shown in the macrocell diagrams.

Eight synchronous preset terms are distributed in a 2/4 pattern. The first four macrocells share Preset 0, the next two share Preset 1, and so on, ending with the last two macrocells sharing Preset 7.

The 14 dedicated inputs and their complements use the numbered positions in the global bus as shown. Each macrocell provides six inputs to the global bus: (left to right) flip-flop Q2 true and false, flip-flop Q1 true and false, and the pin true and false. The positions occupied by these signals in the global bus are the six numbers in the bus diagram next to each macrocell.

Absolute Maximum Ratings*

Temperature Under Bias	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-2.0 V to +7.0 V(1)
Voltage on Input Pins with Respect to Ground During Programming	-2.0 V to +14.0 V(1)
Programming Voltage with Respect to Ground	-2.0 V to +14.0 V(1)
Integrated UV Erase Dose	7258 W·sec/cm^2

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

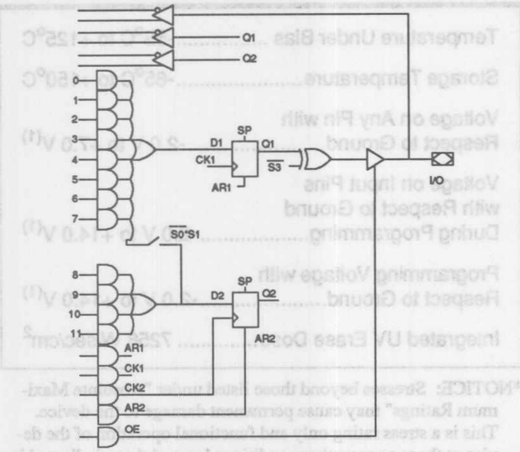
Note:
1. Minimum voltage is -0.6 V dc which may undershoot to -2.0 V for pulses of less than 20 ns. Maximum output pin voltage is V_{CC}+0.75 V dc which may overshoot to +7.0 V for pulses of less than 20 ns.

Operating Modes

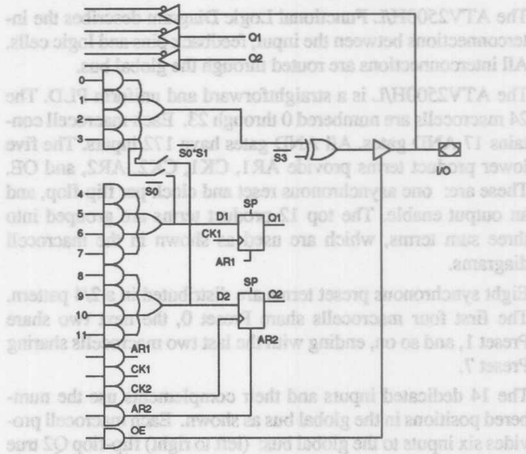
	40-Pin DIP	21	2	38	23	20	V_{CC}(10)	Odd	Even
Mode	44-Pin JLCC	23	2	42	25	22	V_{CC}(11,12)	I/Os	I/Os
"PLD"		X(1)	X	X	X	X	5 V	I/O	I/O
Program		V_{PP}	X	X	X	V_H (2)	6 V	D_{IN}	N.C.
PGM Verify		V_{PP}	X	X	X	V_{IL}	6 V	D_{OUT}	V_{OH}
PGM Inhibit		V_{PP}	X	X	X	V_{IH}	6 V	High Z	High Z
Preload Q1			X	V_H	V_{IL}/V_{IH}	V_{IL}	5 V	D_{IN}(Even/Odd)	V_{IH}
Preload Q2			X	V_H	V_{IL}/V_{IH}	V_{IH}	5 V	D_{IN}(Even/Odd)	V_{IH}
Observe Q2		X	V_H	X	X	X	5 V	D_{OUT}	D_{OUT}

Notes: 1. X can be V_{IL} or V_{IH}.
2. V_H = 11.0 V to 14.0 V

Output Logic, Registered [1]



Output Logic, Combinatorial [1]



Note: 1. These diagrams shows equivalent logic functions, not necessarily the actual circuit implementation.

			Terms In		
S2	S1	S0	D1	D2	Output Configuration
0	0	0	8	4	Registered (Q1)
0	1	0	12	4[1]	Registered (Q1)

Note: 1. These 4 terms are shared with D1.

			Terms In		
S2	S1	S0	D1	D2	Output Configuration
1	0	0	4[1]	4	Combinatorial (8 Terms)
1	0	1	4	4	Combinatorial (4 Terms)
1	1	0	4[1]	4[1]	Combinatorial (12 Terms)

Note: 1. These 4 terms are shared with D1.

S3	Output Configuration
0	Active Low
1	Active High

S3	Output Configuration
0	Active Low
1	Active High

D.C. and A.C. Operating Conditions

		ATV2500H-25	ATV2500H/L-30	ATV2500H/L-35	ATV2500L-40
Operating Temperature (Case)	Com.	0°C - 70°C	0°C - 70°C	0°C - 70°C	0°C - 70°C
	Ind.	-40°C - 85°C	-40°C - 85°C	-40°C - 85°C	-40°C - 85°C
	Mil.	-55°C - 125°C	-55°C - 125°C	-55°C - 125°C	-55°C - 125°C
V_{CC} Power Supply		5 V ± 10%	5 V ± 10%	5 V ± 10%	5 V ± 10%

D.C. Characteristics

Symbol	Parameter	Condition			Min	Typ	Max	Units
I_{LI}	Input Load Current	V_{IN} = -0.1 V to V_{CC}+1 V					10	µA
I_{LO}	Output Leakage Current	V_{OUT} = -0.1 V to V_{CC}+0.1 V					10	µA
I_{CC}	Power Supply Current	V_{CC} = MAX, V_{IN} = GND or V_{CC} Outputs Open	ATV2500L	Com.		0.5	5	mA
				Ind.,Mil.		0.5	10	mA
			ATV2500H	Com.		80	160	mA
				Ind.,Mil.		80	180	mA
I_{CC2}	Clocked Power Supply Current (ATV2500L)	V_{CC} = MAX Outputs Open		Com.		10	15	mA/MHz
				Ind.,Mil.		10	20	mA/MHz
I_{OS} (1)	Output Short Circuit Current	V_{OUT} = 0.5 V					-90	mA
V_{IL}	Input Low Voltage				-0.6		0.8	V
V_{IH}	Input High Voltage				2.0		V_{CC}+0.75	V
V_{OL}	Output Low Voltage	V_{IN} = V_{IH} or V_{IL}, I_{OL} = 8 mA Com,Ind; 6 mA Mil.					0.5	V
V_{OH}	Output High Voltage	I_{OH} = -100 µA			V_{CC}-0.3			V
		I_{OH} = -4.0 mA			2.4			V

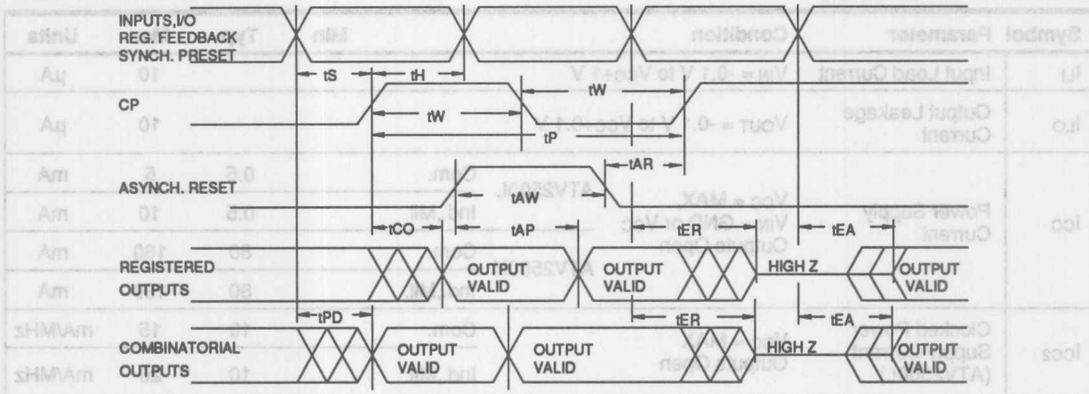
Notes: 1. Not more than one output at a time should be shorted. Duration of short circuit test should not exceed 30 sec.This parameter is only sampled and is not 100% tested. See Absolute Maximum Ratings.

Pin Capacitance (f = 1 MHz, T = 25°C) (1)

	Typ	Max	Units	Conditions
C_{IN}	4	6	pF	V_{IN} = 0 V
C_{OUT}	8	12	pF	V_{OUT} = 0 V

Note: 1. Typical values for nominal supply voltage. This parameter is only sampled and is not 100% tested.

A.C. Waveforms [1]



Note: 1. Timing measurement reference is 1.5 V. Input AC driving levels are 0.0 V and 3.0 V, unless otherwise specified.

A.C. Characteristics for the ATV2500L

Symbol	Parameter	ATV2500L-30		ATV2500L-35		ATV2500L-40		Units
		Min	Max	Min	Max	Min	Max	
t_{PD}	Input or Feedback to Non-Registered Output		30		35		40	ns
t_{EA}	Input to Output Enable		30		35		40	ns
t_{ER}	Input to Output Disable		30		35		40	ns
t_{CO}	Clock to Output	5	30	5	35	5	40	ns
t_{CF}	Clock to Feedback	10	20	15	20	15	22	ns
t_{SI1}	Input Setup Time, Output Register	20		22		25		ns
t_{SI2}	Input Setup Time, Buried Register[1]	20		22		25		ns
t_{SF}	Feedback Setup Time	10		15		18		ns
t_{H1}	Hold Time, Output Register	10		15		15		ns
t_{H2}	Hold Time, Buried Register[1]	5		5		5		ns
t_W	Clock Width	15		17		19		ns
t_P	Clock Period	30		35		40		ns
F_{MAX}	Maximum Frequency ($1/t_P$)		33		28		25	MHz
t_{AW}	Asynchronous Reset Width	18		20		22		ns
t_{AR}	Asynchronous Reset Recovery Time	18		20		22		ns
t_{AP}	Asynchronous Reset to Registered Output Reset		30		35		40	ns

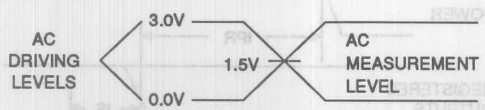
Note: 1. Buried registers include all 24 Q2 registers and any of the 24 Q1 registers in macrocells configured as combinatorial.

A.C. Characteristics for the ATV2500H

Symbol	Parameter	ATV2500H-25 Min	ATV2500H-25 Max	ATV2500H-30 Min	ATV2500H-30 Max	ATV2500H-35 Min	ATV2500H-35 Max	Units
t_{PD}	Input or Feedback to Non-Registered Output		25		30		35	ns
t_{EA}	Input to Output Enable		25		30		35	ns
t_{ER}	Input to Output Disable		25		30		35	ns
t_{CO}	Clock to Output	10	25	12	30	15	35	ns
t_{CF}	Clock to Feedback	10	18	12	20	15	20	ns
t_{SI1}	Input Setup Time, Output Register	10		12		15		ns
t_{SI2}	Input Setup Time, Buried Register[1]	5		5		5		ns
t_{SF}	Feedback Setup Time	7		10		15		ns
t_{H1}	Hold Time	5		5		5		ns
t_W	Clock Width	10		12		15		ns
t_P	Clock Period	25		30		35		ns
F_{MAX}	Maximum Frequency ($1/t_P$)		40		33		28	MHz
t_{AW}	Asynchronous Reset Width	15		18		20		ns
t_{AR}	Asynchronous Reset Recovery Time	15		18		20		ns
t_{AP}	Asynchronous Reset to Registered Output Reset		25		30		35	ns

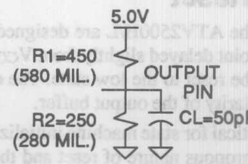
Note: 1. Buried registers include all 24 Q2 registers and any of the 24 Q1 registers in macrocells configured as combinatorial.

Input Test Waveforms and Measurement Levels



t_R, t_F < 5 ns (10% to 90%)

Output Test Load

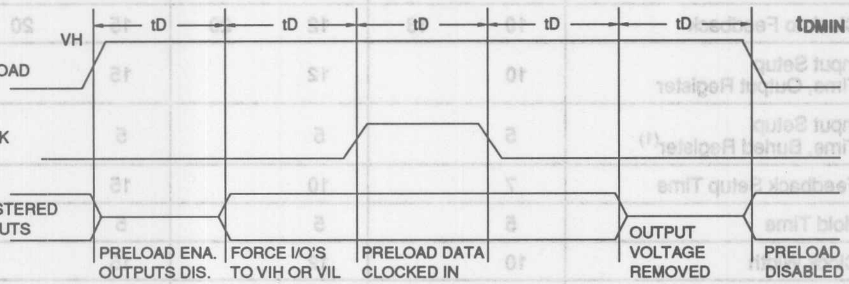


Preload and Observability of Registered Outputs

The ATV2500H/L's registers are provided with circuitry to allow loading of each register asynchronously with either a high or a low. This feature will simplify testing since any state can be forced into the registers to control test sequencing. A V_{IH} level on the Odd I/O pins will force the appropriate register high; a V_{IL} will force it low, independent of the polarity or other configuration bit settings.

The preload state is entered by placing an 11 V to 14 V signal on pin 38 on the DIP and pin 42 on the SMP. When the clock term is pulsed high, (pin 21 on the DIP, pin 23 on the SMP) the data on the I/O pins is placed into the 12 registers chosen by the Q select and even/odd select pins.

Register 2 observability mode is entered by placing an 11 V to 14 V signal on pin 2 (DIP or SMP). In this mode, the contents of the buried register bank will appear on the associated outputs when the OE control signals are active.



Level forced on Odd I/O pin during preload cycle.	Q Select pin state	Even/ Odd select	Even Q1 state after cycle	Even Q2 state after cycle	Odd Q1 state after cycle	Odd Q2 state after cycle
V_{IH}	Low	Low	High	X	X	X
V_{IL}	Low	Low	Low	X	X	X
V_{IH}	High	Low	X	High	X	X
V_{IL}	High	Low	X	Low	X	X
V_{IH}	Low	High	X	X	High	X
V_{IL}	Low	High	X	X	Low	X
V_{IH}	High	High	X	X	X	High
V_{IL}	High	High	X	X	X	Low

Power Up Reset

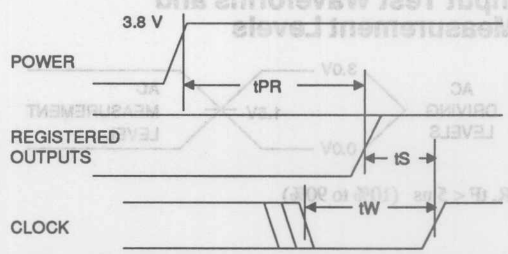
The registers in the ATV2500H/L are designed to reset during power up. At a point delayed slightly from V_{CC} crossing 3.8 V, all registers will be reset to the low state. The output state will depend on the polarity of the output buffer.

This feature is critical for state machine initialization. However, due to the asynchronous nature of reset and the uncertainty of how V_{CC} actually rises in the system, the following conditions are required:

1) The V_{CC} rise must be monotonic,

2) After reset occurs, all input and feedback setup times must be met before driving the clock term high, and

3) The signals from which the clock is derived must remain stable during t_{PR}.



Parameter	Description	Min	Typ	Max	Units
t_{PR}	Power-Up Reset Time		600	1000	ns

Security Fuse Usage

A single fuse is provided to prevent unauthorized copying of the ATV2500H/L fuse patterns. Once programmed, the outputs will read programmed during verify. The security fuse should be programmed last, as its effect is immediate.

The security fuse also inhibits preload and Q2 observability.

Atmel CMOS PLDs

Atmel's Erasable Programmable Logic Devices utilize an advanced 1.25-micron CMOS EPROM technology. This technology's state of the art features are the optimum combination for PLDs:

- CMOS technology provides high speed, low power, and high noise immunity.
- EPROM technology is the most cost effective method for producing PLDs - surpassing bipolar fusible link technology in low cost, while providing the necessary reprogrammability.
- EPROM reprogrammability, which is 100% tested before shipment, provides inherently better programmability and reliability than one-time fusible PLDs.
- Atmel's EPROM process has proven extremely reliable in the volume production of a full line of advanced EPROM memory products, from 64K to one-megabit devices.

Using the ATV2500H/L's Many Advanced Features

The ATV2500H/L's flexibility puts more usable gates in 40 pins than other PLDs. Some of the ATV2500H/L's key features are:

- Asynchronous Clocks -

Each of the flip-flops in the ATV2500H/L has a dedicated product term driving the clock. The user is no longer constrained to using one clock for all the registers. Buried state machines, counters, and registers can all coexist in one device, while running on separate clocks. The ATV2500H/L clock period matches that of similar synchronous devices.

- A Total of 48 Registers -

The ATV2500H/L provides two flip-flops for each output macrocell - a total of 48. Each register has its own clock and reset product terms, as well as its own sum term.

- Independent I/O Pin and Feedback Paths -

Each I/O pin on the ATV2500/H has a dedicated input path. Each of the 48 registers has individual feedback terms into the array. This feature, combined with individual product terms for each I/O's output enable, facilitates designs using bi-directional I/O buses.

- Three Sum Terms per Macrocell -

The ATV2500H/L macrocell can be configured with one SUM term feeding the output, and still have two SUM terms feeding the flip-flops. This is the simplest method for interfacing with an I/O bus, and no flip-flops need be sacrificed.

- Combinable Sum Terms -

Each output macrocell's three SUM terms can be combined in an OR gate before the output or the register. This provides up to 12 product terms per output or flip-flop. When the registered output configuration is chosen, eight terms are automatically available to D1. The four terms feeding D2 can also be shared with D1, giving it a total of 12. In the combinatorial mode, four, eight, or 12 terms can feed the output, with the middle four still driving D1 and the bottom four still driving D2.

Programming Software Support

Software which is capable of transforming Boolean equations, state machine descriptions and truth tables into JEDEC files for the ATV2500H/L is currently available from the following sources:

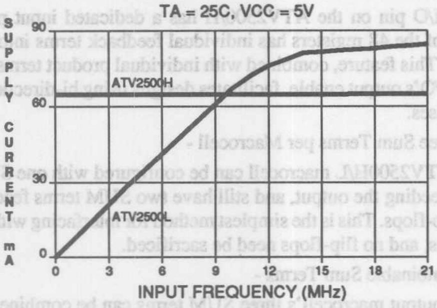
Data I/O Corp.	- Abel™ 4.0 and above
Logical Devices	- Cupl 3.0 and above
Atmel Corp.	-Atmel-Abel-4

Erasure Characteristics

The entire memory array of an ATV2500H/L is erased after exposure to ultraviolet light at a wavelength of 2537 Å. Complete erasure is assured after a minimum of 20 minutes exposure using 12,000 $\mu W/cm^2$ intensity lamps spaced one inch away from the chip. Minimum erase time for lamps at other intensity ratings can be calculated from the minimum integrated erasure dose of 15 $W{\cdot}sec/cm^2$. To prevent unintentional erasure, an opaque label is recommended to cover the clear window on any UV erasable PLD which will be subjected to continuous fluorescent indoor lighting or sunlight.

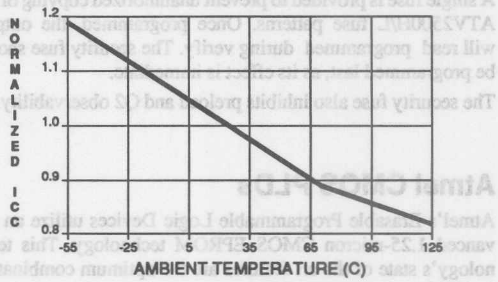
SUPPLY CURRENT vs. INPUT FREQUENCY

TA = 25C, VCC = 5V



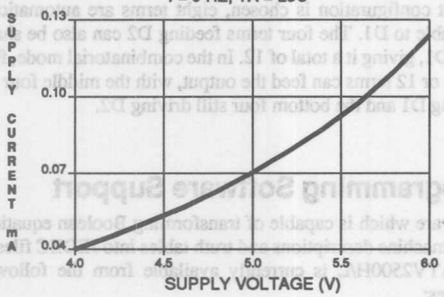
NORMALIZED ICC vs. AMBIENT TEMP.

f = 20 MHz, VCC = 5V



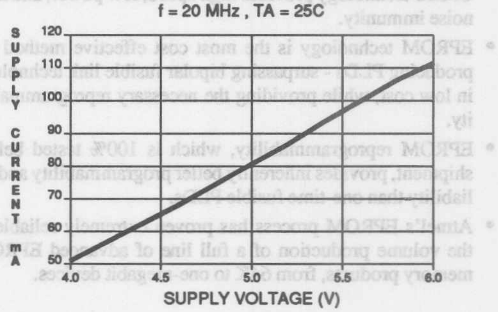
SUPPLY CURRENT vs. SUPPLY VOLTAGE

f = 0 HZ, TA = 25C



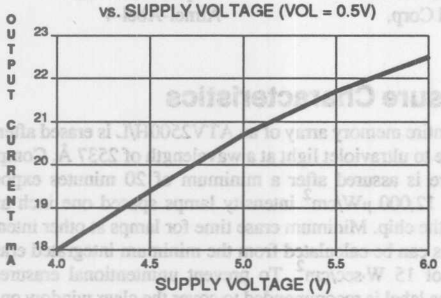
SUPPLY CURRENT vs. SUPPLY VOLTAGE

f = 20 MHz , TA = 25C



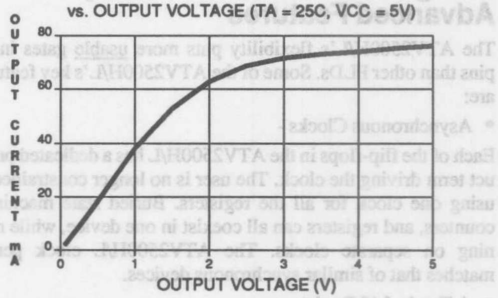
OUTPUT SINK CURRENT

vs. SUPPLY VOLTAGE (VOL = 0.5V)



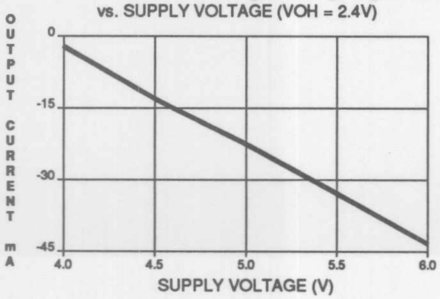
OUTPUT SINK CURRENT

vs. OUTPUT VOLTAGE (TA = 25C, VCC = 5V)



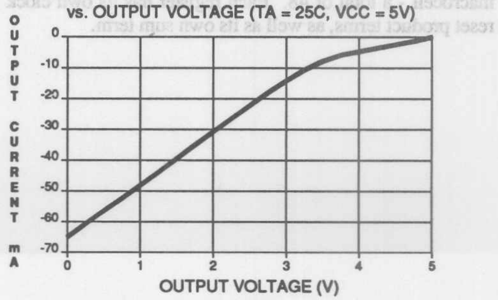
OUTPUT SOURCE CURRENT

vs. SUPPLY VOLTAGE (VOH = 2.4V)

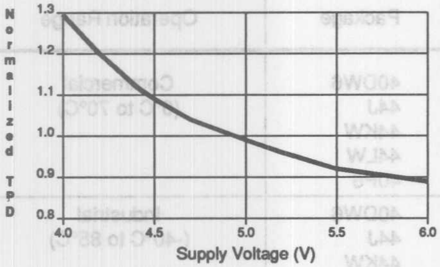


OUTPUT SOURCE CURRENT

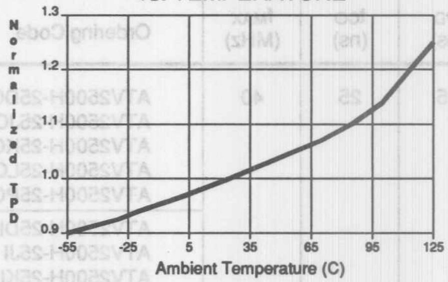
vs. OUTPUT VOLTAGE (TA = 25C, VCC = 5V)



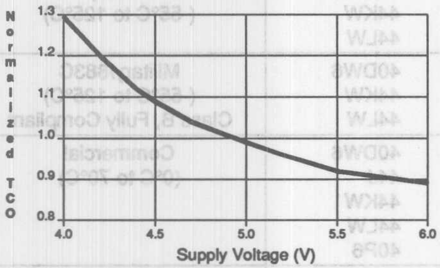
NORMALIZED TPD vs. SUPPLY VOLTAGE



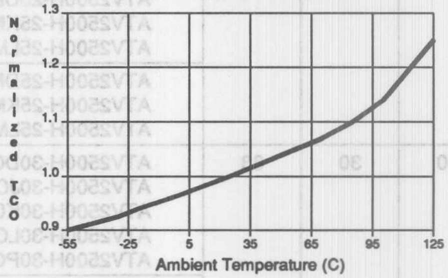
NORMALIZED TPD vs. TEMPERATURE



NORMALIZED TCO vs. SUPPLY VOLTAGE

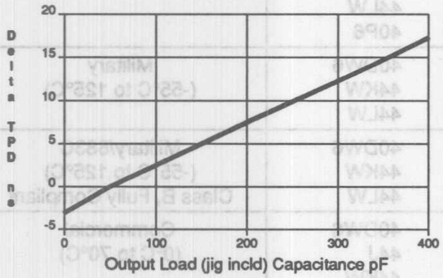


NORMALIZED TCO vs. TEMPERATURE



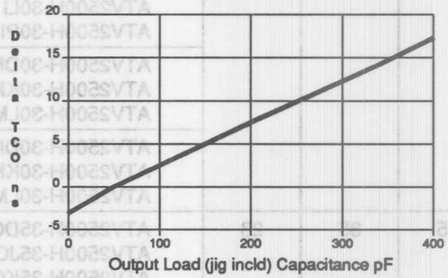
DELTA TPD vs. OUTPUT LOADING

TA = 25C, VCC = 5V



DELTA TCO vs. OUTPUT LOADING

TA = 25C, VCC = 5V



Ordering Information

t_{PD} (ns)	t_{CO} (ns)	f_{MAX} (MHz)	Ordering Code	Package	Operation Range
25	25	40	ATV2500H-25DC ATV2500H-25JC ATV2500H-25KC ATV2500H-25LC ATV2500H-25PC	40DW6 44J 44KW 44LW 40P6	Commercial (0°C to 70°C)
			ATV2500H-25DI ATV2500H-25JI ATV2500H-25KI ATV2500H-25LI ATV2500H-25PI	40DW6 44J 44KW 44LW 40P6	Industrial (-40°C to 85°C)
			ATV2500H-25DM ATV2500H-25KM ATV2500H-25LM	40DW6 44KW 44LW	Military (-55°C to 125°C)
			ATV2500H-25DM/883 ATV2500H-25KM/883 ATV2500H-25LM/883	40DW6 44KW 44LW	Military/883C (-55°C to 125°C) Class B, Fully Compliant
30	30	33	ATV2500H-30DC ATV2500H-30JC ATV2500H-30KC ATV2500H-30LC ATV2500H-30PC	40DW6 44J 44KW 44LW 40P6	Commercial (0°C to 70°C)
			ATV2500H-30DI ATV2500H-30JI ATV2500H-30KI ATV2500H-30LI ATV2500H-30PI	40DW6 44J 44KW 44LW 40P6	Industrial (-40°C to 85°C)
			ATV2500H-30DM ATV2500H-30KM ATV2500H-30LM	40DW6 44KW 44LW	Military (-55°C to 125°C)
			ATV2500H-30DM/883 ATV2500H-30KM/883 ATV2500H-30LM/883	40DW6 44KW 44LW	Military/883C (-55°C to 125°C) Class B, Fully Compliant
35	35	28	ATV2500H-35DC ATV2500H-35JC ATV2500H-35KC ATV2500H-35LC ATV2500H-35PC	40DW6 44J 44KW 44LW 40P6	Commercial (0°C to 70°C)
			ATV2500H-35DI ATV2500H-35JI ATV2500H-35KI ATV2500H-35LI ATV2500H-35PI	40DW6 44J 44KW 44LW 40P6	Industrial (-40°C to 85°C)
			ATV2500H-35DM ATV2500H-35KM ATV2500H-35LM	40DW6 44KW 44LW	Military (-55°C to 125°C)
			ATV2500H-35DM/883 ATV2500H-35KM/883 ATV2500H-35LM/883	40DW6 44KW 44LW	Military/883C (-55°C to 125°C) Class B, Fully Compliant

1

Ordering Information

t_{PD} (ns)	t_{CO} (ns)	f_{MAX} (MHz)	Ordering Code	Package	Operation Range
25	25	40	5962-91545 02M QX 5962-91545 02M XX 5962-91545 02M YX	40DW6 44LW 44KW	Military/833C (-55°C to 125°C) Class B, Fully Compliant
30	30	33	5962-91545 01M QX 5962-91545 01M XX 5962-91545 01M YX	40DW6 44LW 44KW	Military/833C (-55°C to 125°C) Class B, Fully Compliant

Ordering Information

t_{PD} (ns)	t_{CO} (ns)	f_{MAX} (MHz)	Ordering Code	Package	Operation Range
30	30	33	ATV2500L-30DC	40DW6	Commercial (0°C to 70°C)
			ATV2500L-30JC	44J	
			ATV2500L-30KC	44KW	
			ATV2500L-30LC	44LW	
			ATV2500L-30PC	40P6	
			ATV2500L-30DI	40DW6	Industrial (-40°C to 85°C)
			ATV2500L-30JI	44J	
			ATV2500L-30KI	44KW	
			ATV2500L-30LI	44LW	
			ATV2500L-30PI	40P6	
			ATV2500L-30DM	40DW6	Military (-55°C to 125°C)
			ATV2500L-30KM	44KW	
			ATV2500L-30LM	44LW	
			ATV2500L-30DM/883	40DW6	Military (-55°C to 125°C) Class B, Fully Compliant
			ATV2500L-30KM/883	44KW	
			ATV2500L-30LM/883	44LW	
35	35	28	ATV2500L-35DC	40DW6	Commercial (0°C to 70°C)
			ATV2500L-35JC	44J	
			ATV2500L-35KC	44KW	
			ATV2500L-35LC	44LW	
			ATV2500L-35PC	40P6	
			ATV2500L-35DI	40DW6	Industrial (-40°C to 85°C)
			ATV2500L-35JI	44J	
			ATV2500L-35KI	44KW	
			ATV2500L-35LI	44LW	
			ATV2500L-35PI	40P6	
			ATV2500L-35DM	40DW6	Military (-55°C to 125°C)
			ATV2500L-35KM	44KW	
			ATV2500L-35LM	44LW	
			ATV2500L-35DM/883	40DW6	Military (-55°C to 125°C) Class B, Fully Compliant
			ATV2500L-35KM/883	44KW	
			ATV2500L-35LM/883	44LW	
40	40	25	ATV2500L-40DC	40DW6	Commercial (0°C to 70°C)
			ATV2500L-40JC	44J	
			ATV2500L-40KC	44KW	
			ATV2500L-40LC	44LW	
			ATV2500L-40PC	40P6	
			ATV2500L-40DI	40DW6	Industrial (-40°C to 85°C)
			ATV2500L-40JI	44J	
			ATV2500L-40KI	44KW	
			ATV2500L-40LI	44LW	
			ATV2500L-40PI	40P6	
			ATV2500L-40DM	40DW6	Military (-55°C to 125°C)
			ATV2500L-40KM	44KW	
			ATV2500L-40LM	44LW	

1

Ordering Information

t_{PD} (ns)	t_{CO} (ns)	f_{MAX} (MHz)	Ordering Code	Package	Operation Range
40	40	25	ATV2500L-40DM/883 ATV2500L-40KM/883 ATV2500L-40LM/883	40DW6 44KW 44LW	Military/883C (-55°C to 125°C) Class B, Fully Compliant
30	30	33	5962-91545 03M QX 5962-91545 03M XX 5962-91545 03M YX	40DW6 44LW 44KW	Military/833C (-55°C to 125°C) Class B, Fully Compliant

Package Type	
40DW6	40 Lead, 0.600" Wide Windowed, Ceramic Dual Inline Package (Cerdip)
44J	44 Lead, Plastic J-Leaded Chip Carrier OTP (PLCC)
44KW	44 Lead, Windowed, Ceramic J-Leaded Chip Carrier (JLCC)
44LW	44 Pad, Windowed, Ceramic Leadless Chip Carrier (LCC)
40P6	40 Lead, 0.600" Wide Plastic Dual Inline Package OTP (PDIP)

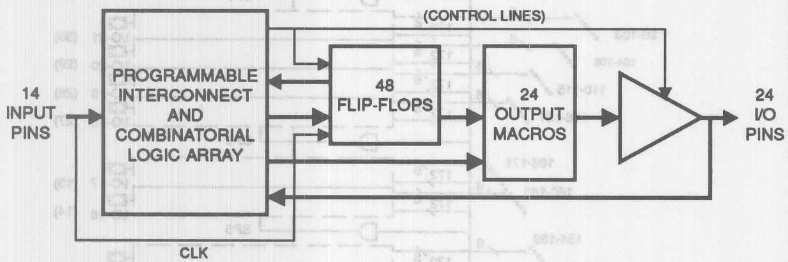
1

High Speed High Density UV Erasable Programmable Logic Device

Features

- High Performance, High Density Programmable Logic Device
  - Typical 7 ns Pin-to-Pin Delay
  - Fully Connected Logic Array With 416 Product Terms
- Flexible Output Macrocell
  - 48 Flip-Flops - Two per Macrocell
  - 72 Sum Terms
  - All Flip-Flops, I/O Pins Feed In Independently
  - Achieves Over 80% Gate Utilization
- Enhanced Macrocell Configuration Selections
  - D- or T-Type Flip-Flops
  - Product Term or Direct Input Pin Clocking
  - Registered or Combinatorial Internal Feedback
- Low Power ATV2500BL - 2 mA Stand-By (Typical)
- Backward Compatible With ATV2500H/L Software
- Proven and Reliable High Speed UV EPROM Process
- Reprogrammable - Tested 100% for Programmability
- 44-Lead Surface Mount Packages

Block Diagram



Description

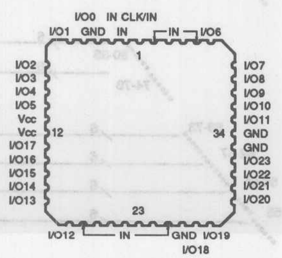
The ATV2500Bs are the highest density PLDs available in a 44-pin package. With their fully connected logic array and flexible macrocell structure, high gate utilization is easily obtainable.

The ATV2500Bs are organized around a *single universal input bus*. All pin and feedback terms are always available to every macrocell. Each of the 38 logic pins are array inputs, as are the outputs of each flip-flop. *(continued)*

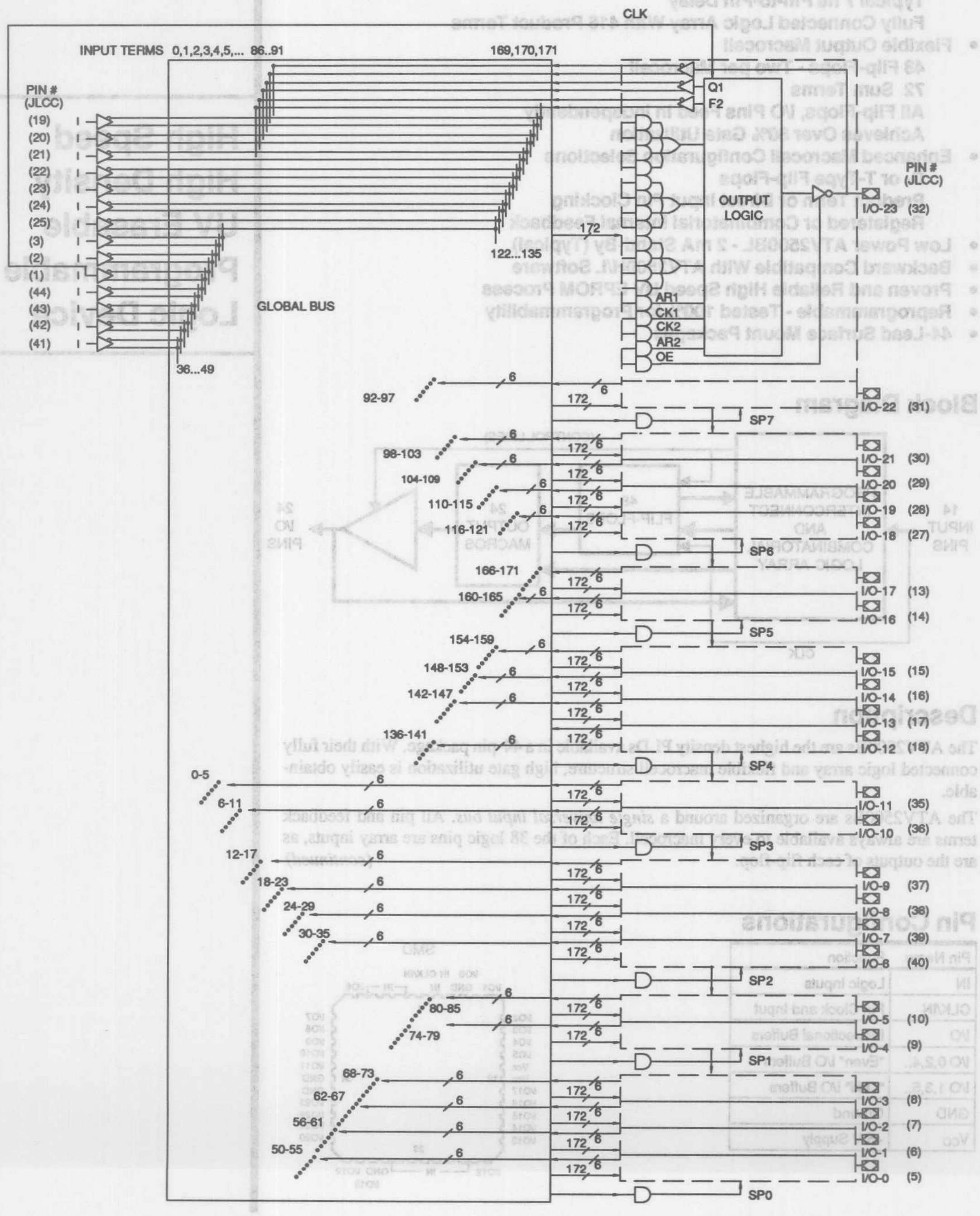
Pin Configurations

Pin Name	Function
IN	Logic Inputs
CLK/IN	Pin Clock and Input
I/O	Bidirectional Buffers
I/O 0,2,4..	"Even" I/O Buffers
I/O 1,3,5..	"Odd" I/O Buffers
GND	Ground
Vcc	+5 V Supply

SMD



Functional Logic Diagram ATV2500B



Description (Continued)

In the ATV2500Bs, four product terms are input to each sum term. Furthermore, each macrocell's three sum terms can be combined to provide up to 12 product terms per sum term with *no performance penalty*. Each flip-flop is individually selectable to be either D- or T-type, providing further logic compaction. Also, 24 of the flip-flops may be bypassed to provide internal combinatorial feedback to the logic array.

Product terms provide individual clocks and asynchronous resets for each flip-flop. The flip-flops may also be individually configured to have direct input pin clocking. Each output has its own enable product term. Eight synchronous preset product terms serve local groups of either four or eight flip-flops. Register preload functions are provided to simplify testing. All registers automatically reset upon power up.

The ATV2500BL is the low voltage compatible device with speeds as fast as 15 ns. The ATV2500BL consumes only 2 mA at standby, which provides the optimum low power PLD solution with full CMOS output levels. The ATV2500BL significantly reduces total system power, allowing battery-powered operation.

Functional Logic Diagram Description

The ATV2500B functional logic diagram describes the interconnections between the input, feedback pins and logic cells. All interconnections are routed through the single global bus.

The ATV2500Bs are straightforward and uniform PLDs. The 24 macrocells are numbered 0 through 23. Each macrocell contains 17 AND gates. All AND gates have 172 inputs. The five lower product terms provide AR1, CK1, CK2, AR2, and OE. These are: one asynchronous reset and clock per flip-flop, and an output enable. The top 12 product terms are grouped into three sum terms, which are used as shown in the macrocell diagrams.

Eight synchronous preset terms are distributed in a 2/4 pattern. The first four macrocells share Preset 0, the next two share Preset 1, and so on, ending with the last two macrocells sharing Preset 7.

The 14 dedicated inputs and their complements use the numbered positions in the global bus as shown. Each macrocell provides six inputs to the global bus: (left to right) feedback F2[1] true and false, flip-flop Q1 true and false, and the pin true and false. The positions occupied by these signals in the global bus are the six numbers in the bus diagram next to each macrocell.

Note: 1. Either the flip-flop input (D/T2) or output (Q2) may be fed back in the ATV2500Bs.

Operating Modes

Mode	44-Lead SMD	23	2	42	25	22	V_{CC}(11,12)	Odd I/Os	Even I/Os
"PLD"		X[1]	X	X	X	X	4.5 V - 5.5 V	I/O	I/O
Program		V_{PP}	X	X	X	V_H [2]	6 V	D_{IN}	N.C.
PGM Verify		V_{PP}	X	X	X	V_{IL}	6 V	D_{OUT}	V_{OH}
PGM Inhibit		V_{PP}	X	X	X	V_{IH}	6 V	High Z	High Z
Preload Q1			X	V_H	V_{IL}/V_{IH}	V_{IL}	4.5 V - 5.5 V	D_{IN}(Even/Odd)	V_{IH}
Preload Q2			X	V_H	V_{IL}/V_{IH}	V_{IH}	4.5 V - 5.5 V	D_{IN}(Even/Odd)	V_{IH}
Observe Q2		X	V_H	X	X	X	4.5 V - 5.5 V	D_{OUT}	D_{OUT}

Notes: 1. X can be V_{IL} or V_{IH}.
2. V_H = 10.25 V to 10.75 V.

Absolute Maximum Ratings*

Temperature Under Bias	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-2.0 V to +7.0 V(1)
Voltage on Input Pins with Respect to Ground During Programming	-2.0 V to +14.0 V(1)
Programming Voltage with Respect to Ground	-2.0 V to +14.0 V(1)
Integrated UV Erase Dose	7258 W·sec/cm²

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Note:
1. Minimum voltage is -0.6 V dc which may undershoot to -2.0 V for pulses of less than 20 ns. Maximum output pin voltage is V_{CC}+0.75 V dc which may overshoot to +7.0 V for pulses of less than 20 ns.

D.C. and A.C. Operating Conditions

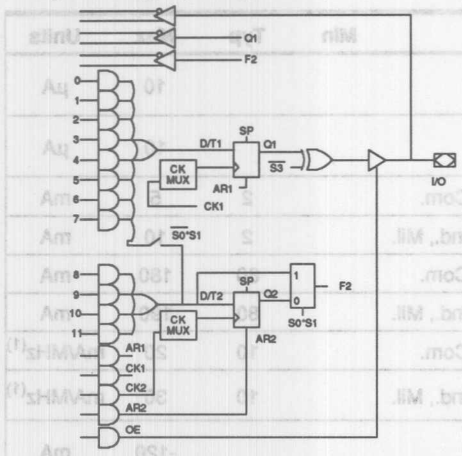
	Commercial	Industrial	Military
Operating Temperature (Case)	0°C - 70°C	-40°C - 85°C	-55°C - 125°C
V_{CC} Power Supply	5 V ± 5%	5 V ± 10%	5 V ± 10%

Pin Capacitance (f = 1 MHz, T = 25°C) (1)

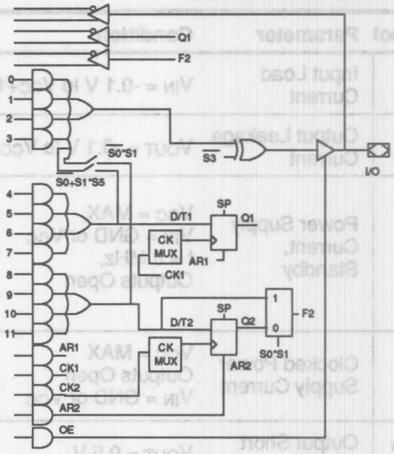
	Typ	Max	Units	Conditions
C_{IN}	4	6	pF	V_{IN} = 0 V
C_{OUT}	8	12	pF	V_{OUT} = 0 V

Note: 1. Typical values for nominal supply voltage. This parameter is only sampled and is not 100% tested.

Output Logic, Registered [1]



Output Logic, Combinatorial [1]



Note: 1. These diagrams show equivalent logic functions, not necessarily the actual circuit implementation.

S2 = 0		Terms In		
S1	S0	D/T1	D/T2	Output Configuration
0	0	8	4	Registered (Q1); Q2 FB
1	0	12	4[1]	Registered (Q1); Q2 FB
1	1	8	4	Registered (Q1); D/T2 FB

Note: 1. These four terms are shared with D/T1.

S2 = 1			Terms In		
S5	S1	S0	D/T1	D/T2	Output Configuration
X	0	0	4[1]	4	Combinatorial (8 Terms); Q2 FB
X	0	1	4	4	Combinatorial (4 Terms); Q2 FB
X	1	0	4[1]	4[1]	Combinatorial (12 Terms); Q2 FB
1	1	1	4[1]	4	Combinatorial (8 Terms); D/T2 FB
0	1	1	4	4	Combinatorial (4 Terms); D/T2 FB

Note: 1. These four terms are shared with D/T1.

S3	Output Configuration
0	Active Low
1	Active High

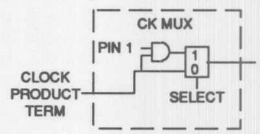
S6	Q1 CLOCK
0	CK1
1	CK1 • PIN1

S4	Register 1 Type
0	D
1	T

S7	Q2 CLOCK
0	CK2
1	CK2 • PIN1

S5	Register 2 Type
0	D
1	T

Clock Option

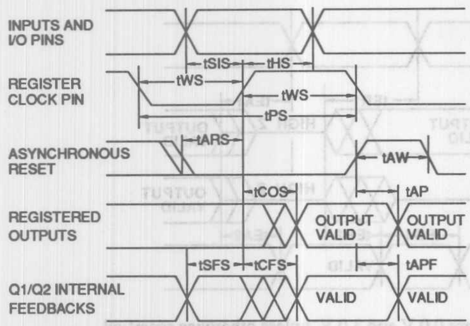


D.C. Characteristics

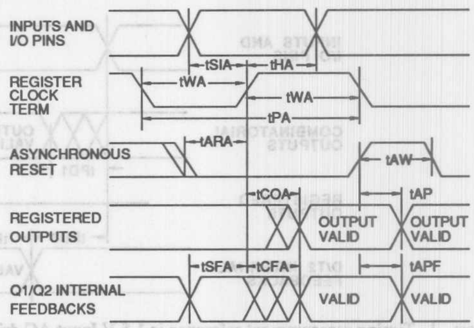
Symbol	Parameter	Condition			Min	Typ	Max	Units
I_{LI}	Input Load Current	V_{IN} = -0.1 V to V_{CC}+1 V					10	µA
I_{LO}	Output Leakage Current	V_{OUT} = -0.1 V to V_{CC}+0.1 V					10	µA
I_{CC}	Power Supply Current, Standby	V_{CC} = MAX, V_{IN} = GND or V_{CC} f = 0 MHz, Outputs Open	ATV2500BL	Com.		2	5	mA
				Ind., Mil.		2	10	mA
			ATV2500B	Com.		80	180	mA
				Ind., Mil.		80	190	mA
I_{CC2}	Clocked Power Supply Current	V_{CC} = MAX Outputs Open V_{IN} = GND or V_{CC}	ATV2500BL	Com.		10	20	mA/MHz[1]
				Ind., Mil.		10	30	mA/MHz[1]
I_{OS} [2]	Output Short Circuit Current	V_{OUT} = 0.5 V					-120	mA
V_{IL}	Input Low Voltage	MIN ≤ V_{CC} ≤ MAX			-0.6		0.8	V
V_{IH}	Input High Voltage				2.0		V_{CC}+0.75	V
V_{OL}	Output Low Voltage	V_{IN} = V_{IH} or V_{IL}, V_{CC} = 4.5 V;	I_{OL} = 8 mA Com,Ind				0.5	V
			I_{OL} = 6 mA Mil.				0.5	V
V_{OH}	Output High Voltage	V_{CC} = MIN	I_{OH} = -100 µA		V_{CC}-0.3			V
			I_{OH} = -4.0 mA		2.4			V

Notes: 1. See Icc versus frequency characterization curves.
2. Not more than one output at a time should be shorted. Duration of short circuit test should not exceed 30 sec. This parameter is only sampled and is not 100% tested. See Absolute Maximum Ratings.

A.C. Waveforms [1] Input Pin Clock



A.C. Waveforms [1] Product Term Clock



Notes: 1. Timing measurement reference is 1.5 V Input AC driving levels are 0.0 V and 3.0 V, unless otherwise specified.

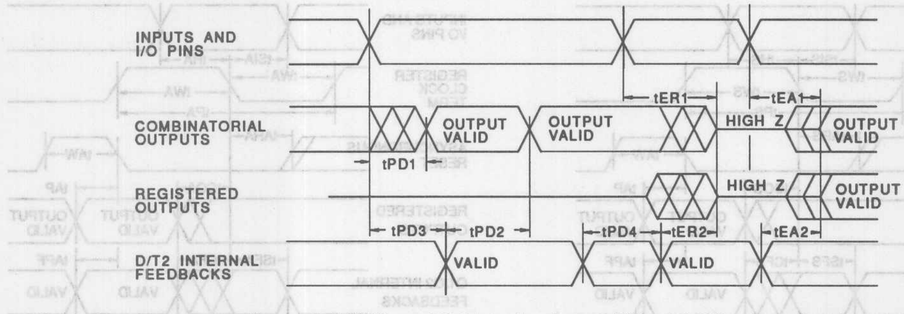
Register A.C. Characteristics, Input Pin Clock

Symbol	Parameter	-12 Min	-12 Max	-15 Min	-15 Max	-20 Min	-20 Max	Units
t_{COS}	Clock to Output		7		10		13	ns
t_{CFS}	Clock to Feedback	0	4	0	5	0	7	ns
t_{SIS}	Input Setup Time	7		9		11		ns
t_{SFS}	Feedback Setup Time	7		9		11		ns
t_{HS}	Hold Time	0		0		0		ns
t_{WS}	Clock Width	5		6		7		ns
t_{PS}	Clock Period	10		12		14		ns
F_{MAXS}	External Feedback $1/(t_{SIS} + t_{COS})$		71		52		41	MHz
	Internal Feedback $1/(t_{SFS} + t_{CFS})$		90		71		55	MHz
	No Feedback $1/(t_{PS})$		100		83		71	MHz
t_{ARS}	Asynchronous Reset/Preset Recovery Time	7		12		15		ns

Register A.C. Characteristics, Product Term Clock

Symbol	Parameter	-12 Min	-12 Max	-15 Min	-15 Max	-20 Min	-20 Max	Units
t_{COA}	Clock to Output		10		15		20	ns
t_{CFA}	Clock to Feedback		7	5	12	7	16	ns
t_{SIA}	Input Setup Time	4		5		5		ns
t_{SFA}	Feedback Setup Time	4		5		5		ns
t_{HA}	Hold Time	3		5		5		ns
t_{WA}	Clock Width	5		6		7		ns
t_{PA}	Clock Period	10		12		14		ns
F_{MAXA}	External Feedback $1/(t_{SIA} + t_{COA})$		71		50		40	MHz
	Internal Feedback $1/(t_{SFA} + t_{CFA})$		90		58		47	MHz
	No Feedback $1/(t_{PS})$		100		83		71	MHz
t_{ARA}	Asynchronous Reset/Preset Recovery Time	3		8		12		ns

A.C. Waveforms [1]
Combinatorial Outputs and Feedback

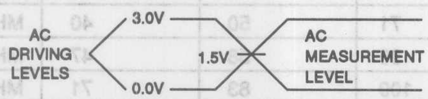


Notes: 1. Timing measurement reference is 1.5 V Input AC driving levels are 0.0 V and 3.0 V, unless otherwise specified.

A.C. Characteristics

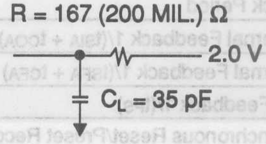
Symbol	Parameter	-12 Min	-12 Max	-15 Min	-15 Max	-20 Min	-20 Max	Units
t_{PD1}	Input to Non-Registered Output		12		15		20	ns
t_{PD2}	Feedback to Non-Registered Output		12		15		20	ns
t_{PD3}	Input to Non-Registered Feedback		7		10		15	ns
t_{PD4}	Feedback to Non-Registered Feedback		7		10		15	ns
t_{EA1}	Input to Output Enable		12		15		20	ns
t_{ER1}	Input to Output Disable		12		15		20	ns
t_{EA2}	Feedback to Output Enable		12		15		20	ns
t_{ER2}	Feedback to Output Disable		12		15		20	ns
t_{AW}	Asynchronous Reset Width	6		8		10		ns
t_{AP}	Asynchronous Reset to Registered Output		15		18		22	ns
t_{APF}	Asynchronous Reset to Registered Feedback		12		15		19	ns

Input Test Waveforms and Measurement Levels



t_R, t_F < 3 ns (10% to 90%)

Output Test Load



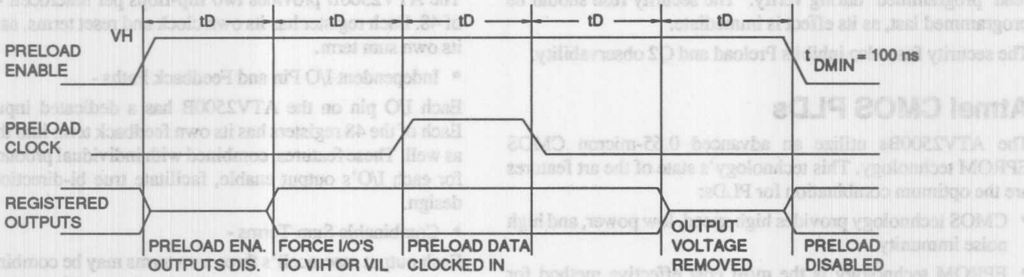
1

Preload and Observability of Registered Outputs

The ATV2500Bs registers are provided with circuitry to allow loading of each register asynchronously with either a high or a low. This feature will simplify testing since any state can be forced into the registers to control test sequencing. A V_{IH} level on the odd I/O pins will force the appropriate register high; a V_{IL} will force it low, independent of the polarity or other configuration bit settings.

The PRELOAD state is entered by placing an 10.25 V to 10.75 V signal on SMP lead 42. When the preload clock SMP lead 23 is pulsed high, the data on the I/O pins is placed into the 12 registers chosen by the Q select and even/odd select pins.

Register 2 observability mode is entered by placing an 10.25 V to 10.75 V signal on pin/lead 2. In this mode, the contents of the buried register bank will appear on the associated outputs when the OE control signals are active.



Level forced on Odd I/O pin during PRELOAD cycle.	Q Select Pin State	Even/ Odd Select	Even Q1 state after cycle	Even Q2 state after cycle	Odd Q1 state after cycle	Odd Q2 state after cycle
V_{IH}	Low	Low	High	X	X	X
V_{IL}	Low	Low	Low	X	X	X
V_{IH}	High	Low	X	High	X	X
V_{IL}	High	Low	X	Low	X	X
V_{IH}	Low	High	X	X	High	X
V_{IL}	Low	High	X	X	Low	X
V_{IH}	High	High	X	X	X	High
V_{IL}	High	High	X	X	X	Low

Power Up Reset

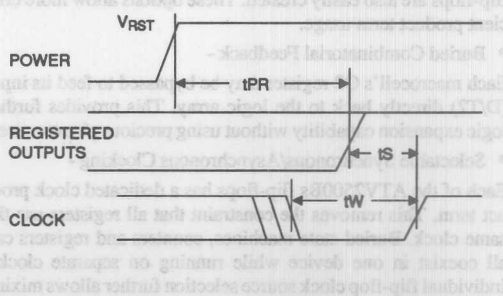
The registers in the ATV2500Bs are designed to reset during power up. At a point delayed slightly from V_{CC} crossing V_{RST}, all registers will be reset to the low state. The output state will depend on the polarity of the output buffer.

This feature is critical for state machine initialization. However, due to the asynchronous nature of reset and the uncertainty of how V_{CC} actually rises in the system, the following conditions are required:

1) The V_{CC} rise must be monotonic,

2) After reset occurs, all input and feedback setup times must be met before driving the clock term high, and

3) The signals from which the clock is derived must remain stable during t_{PR}.



Parameter	Description	Typ	Max	Units
t_{PR}	Power-Up Reset Time	600	1000	ns
V_{RST}	Power-Up Reset Voltage	3.8	4.5	V

Security Fuse Usage

A single fuse is provided to prevent unauthorized copying of ATV2500B fuse patterns. Once programmed, the outputs will read programmed during verify. The security fuse should be programmed last, as its effect is immediate.

The security fuse also inhibits Preload and Q2 observability.

Atmel CMOS PLDs

The ATV2500Bs utilize an advanced 0.65-micron CMOS EPROM technology. This technology's state of the art features are the optimum combination for PLDs:

- CMOS technology provides high speed, low power, and high noise immunity.
- EPROM technology is the most cost effective method for producing PLDs - surpassing bipolar fusible link technology in low cost, while providing the necessary reprogrammability.
- EPROM reprogrammability, which is 100% tested before shipment, provides inherently better programmability and reliability than one-time fusible PLDs.

Using the ATV2500Bs Many Advanced Features

The ATV2500Bs advanced flexibility packs more usable gates into 44 leads than other PLDs. Some of the ATV2500Bs key features are:

- Fully Connected Logic Array -

Each array input is always available to every product term. This makes logic placement a breeze.

- Selectable D- and T-Type Registers -

Each ATV2500B flip-flop can be individually configured as either D- or T-type. Using the T-type configuration, JK and SR flip-flops are also easily created. These options allow more efficient product term usage.

- Buried Combinatorial Feedback -

Each macrocell's Q2 register may be bypassed to feed its input (D/T2) directly back to the logic array. This provides further logic expansion capability without using precious pin resources.

- Selectable Synchronous/Asynchronous Clocking -

Each of the ATV2500Bs flip-flops has a dedicated clock product term. This removes the constraint that all registers use the same clock. Buried state machines, counters and registers can all coexist in one device while running on separate clocks. Individual flip-flop clock source selection further allows mixing higher performance pin clocking and flexible product term clocking within one design.

- A Total of 48 Registers -

The ATV2500B provides two flip-flops per macrocell - a total of 48. Each register has its own clock and reset terms, as well as its own sum term.

- Independent I/O Pin and Feedback Paths -

Each I/O pin on the ATV2500B has a dedicated input path. Each of the 48 registers has its own feedback term into the array as well. These features, combined with individual product terms for each I/O's output enable, facilitate true bi-directional I/O design.

- Combinable Sum Terms -

Each output macrocell's three sum terms may be combined into a single term. This provides a fan in of up to 12 product terms per sum term with *no speed penalty*.

Programming Software Support

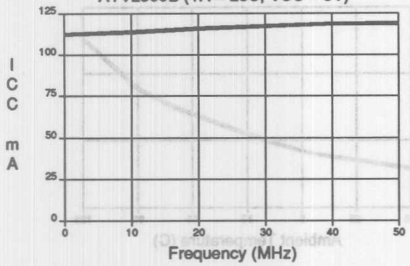
As with all other Atmel PLDs, several third party PLD development software products and programmers will support the ATV2500Bs. Atmel-Abel-4.41, Abel 5.1, CUPL 4.4c, and PLDesigner 3.2 software will fully support all ATV2500B features. Also, any software which supports the ATV2500H/L will automatically support the same features on the ATV2500B. This includes Atmel-Abel-4, Abel 4.0 and above, as well as Cupl 3.0 and above.

Several third party programmers will support the ATV2500B as well. Additionally, the ATV2500B may be programmed to perform the ATV2500H/Ls functional subset (no T-type flip-flops, pin clocking or D/T2 feedback) using the ATV2500H/L JEDEC file. In this case, the ATV2500B becomes a direct replacement or speed upgrade for the ATV2500H/L (additional GND and Vcc connections are required).

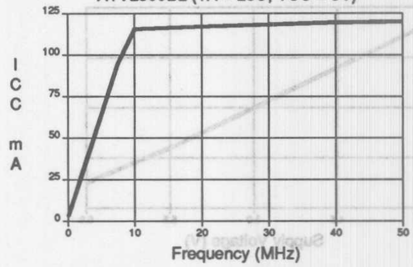
Erasure Characteristics

The entire memory array of an ATV2500B is erased after exposure to ultraviolet light at a wavelength of 2537 Å. Complete erasure is assured after a minimum of 20 minutes exposure using 12,000 $\mu W/cm^2$ intensity lamps spaced one inch away from the chip. Minimum erase time for lamps at other intensity ratings can be calculated from the minimum integrated erasure dose of 15 $W{\cdot}sec/cm^2$. To prevent unintentional erasure, an opaque label is recommended to cover the clear window on any UV erasable PLD which will be subjected to continuous fluorescent indoor lighting or sunlight.

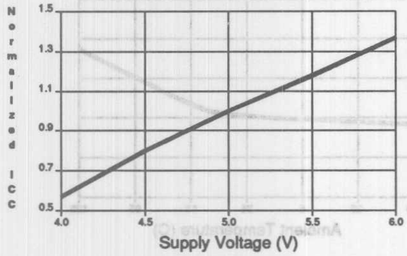
SUPPLY CURRENT vs. INPUT FREQUENCY
ATV2500B (TA = 25C, VCC = 5V)



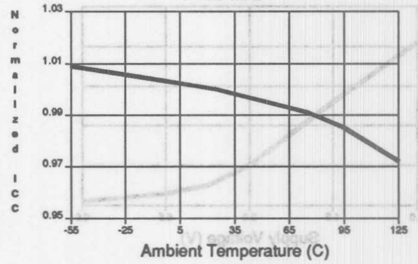
SUPPLY CURRENT vs. INPUT FREQUENCY
ATV2500BL (TA = 25C, VCC = 5V)



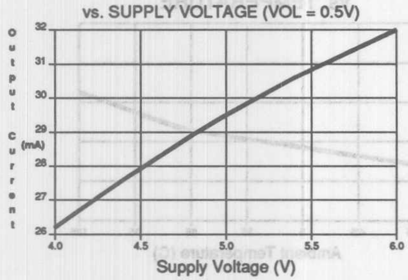
NORMALIZED SUPPLY CURRENT vs. SUPPLY VOLTAGE



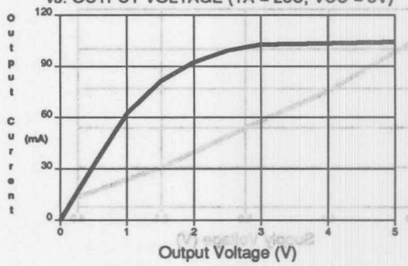
NORMALIZED ICC vs. AMBIENT TEMP.
f = 50 MHz



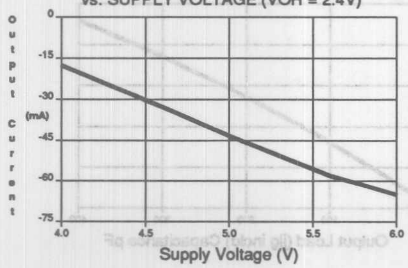
OUTPUT SINK CURRENT
vs. SUPPLY VOLTAGE (VOL = 0.5V)



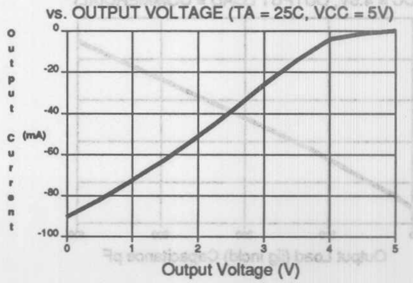
OUTPUT SINK CURRENT
vs. OUTPUT VOLTAGE (TA = 25C, VCC = 5V)



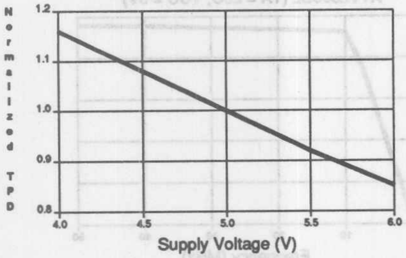
OUTPUT SOURCE CURRENT
vs. SUPPLY VOLTAGE (VOH = 2.4V)



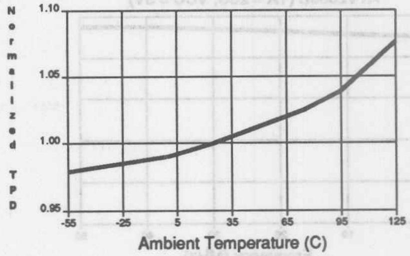
OUTPUT SOURCE CURRENT
vs. OUTPUT VOLTAGE (TA = 25C, VCC = 5V)



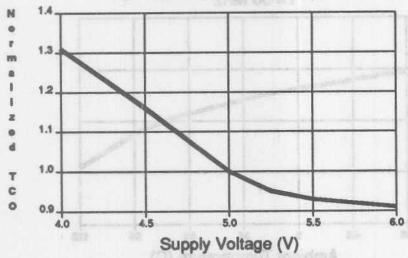
NORMALIZED TPD
vs. SUPPLY VOLTAGE



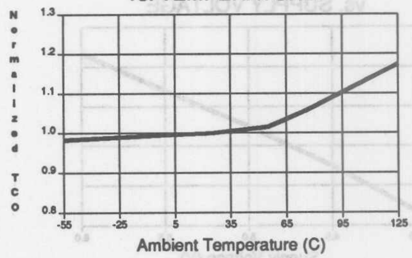
NORMALIZED TPD
vs. TEMPERATURE



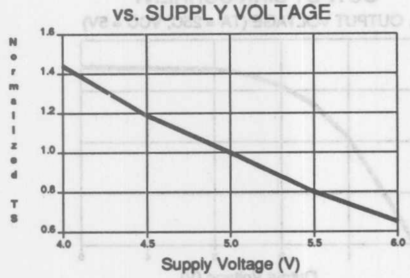
NORMALIZED TCO
vs. SUPPLY VOLTAGE



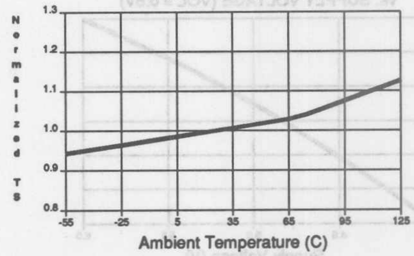
NORMALIZED TCO
vs. TEMPERATURE



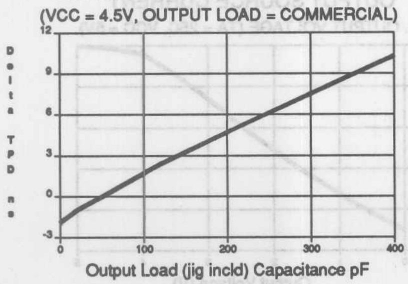
NORMALIZED TS
vs. SUPPLY VOLTAGE



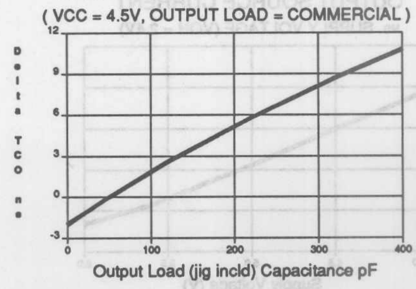
NORMALIZED TS
vs. TEMPERATURE



DELTA TPD vs. OUTPUT LOADING
(VCC = 4.5V, OUTPUT LOAD = COMMERCIAL)



DELTA TCO vs. OUTPUT LOADING
(VCC = 4.5V, OUTPUT LOAD = COMMERCIAL)



Ordering Information

t_{PD} (ns)	t_{COS} (ns)	Ext. f_{MAXS} (MHz)	Ordering Code	Package	Operation Range
12	7	71	ATV2500B-12JC ATV2500B-12KC	44J 44KW	Commercial (0°C to 70°C)
15	10	50	ATV2500B-15JC ATV2500B-15KC	44J 44KW	Commercial (0°C to 70°C)
			ATV2500B-15JI ATV2500B-15KI	44J 44KW	Industrial (-40°C to 85°C)
			ATV2500B-15KM ATV2500B-15LM	44KW 44LW	Military (-55°C to 125°C)
			ATV2500B-15KM/883 ATV2500B-15LM/883	44KW 44LW	Military/883C (-55°C to 125°C) Class B, Fully Compliant
15	10	50	ATV2500BL-15JC ATV2500BL-15KC	44J 44KW	Commercial (0°C to 70°C)
			ATV2500BL-15JI ATV2500BL-15KI	44J 44KW	Industrial (-40°C to 85°C)
20	13	40	ATV2500BL-20JC ATV2500BL-20KC	44J 44KW	Commercial (0°C to 70°C)
			ATV2500BL-20JI ATV2500BL-20KI	44J 44KW	Industrial (-40°C to 85°C)
			ATV2500BL-20KM ATV2500BL-20LM	44KW 44LW	Military (-55°C to 125°C)
			ATV2500BL-20KM/883 ATV2500BL-20LM/883	44KW 44LW	Military/883C (-55°C to 125°C) Class B, Fully Compliant

Package Type	
44J	44 Lead, Plastic J-Leaded Chip Carrier OTP (PLCC)
44KW	44 Lead, Windowed, Ceramic J-Leaded Chip Carrier (JLCC)
44LW	44 Pad, Windowed, Ceramic Leadless Chip Carrier (LCC)

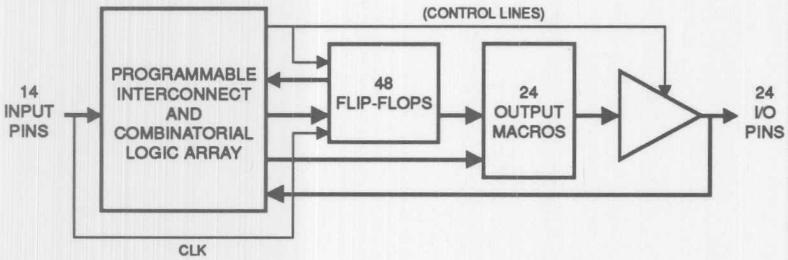
High Speed High Density UV Erasable Programmable Logic Device

Advanced Information

Features

- Quarter Power Equivalent of ATV2500B/BL - 80 mA Maximum
- Low Power ATV2500BQL - 2 mA Standby (Typical)
- High Performance, High Density Programmable Logic Device
  - Maximum 15 ns Pin-to-Pin Delay
  - Fully Connected Logic Array With 416 Product Terms
- Flexible Output Macrocell
  - 48 Flip-Flops - Two per Macrocell
  - 72 Sum Terms
  - All Flip-Flops, I/O Pins Feed In Independently
  - Achieves Over 80% Gate Utilization
- Enhanced Macrocell Configuration Selections
  - D- or T-Type Flip-Flops
  - Product Term or Direct Input Pin Clocking
  - Registered or Combinatorial Internal Feedback
- Backward Compatible With ATV2500H/L Software
- Proven and Reliable High Speed UV EPROM Process
- Reprogrammable - Tested 100% for Programmability
- 40-Pin Dual-In-Line and 44-Lead Surface Mount Packages

Block Diagram



Description

The ATV2500BQs are the highest density PLDs available in a 40-pin package. With their fully connected logic array and flexible macrocell structure, high gate utilization is easily obtainable.

The ATV2500BQs are organized around a *single universal input bus*. All pin and feedback terms are always available to every macrocell. Each of the 38 logic pins are array inputs, as are the outputs of each flip-flop.

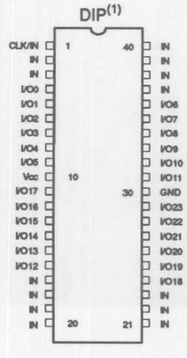
(continued)

Pin Configurations

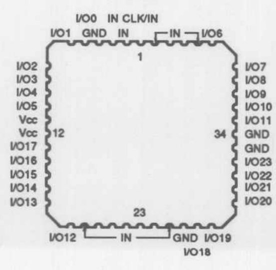
Pin Name	Function
IN	Logic Inputs
CLK/IN	Pin Clock and Input
I/O	Bidirectional Buffers
GND	Ground
Vcc	+5 V Supply

Note: 1. DIP available for -25 and -30 only

DIP[(1)]



SMD



Description (Continued)

In the ATV2500BQs, four product terms are input to each sum term. Furthermore, each macrocell's three sum terms can be combined to provide up to 12 product terms per sum term with *no performance penalty*. Each flip-flop is individually selectable to be either D- or T-type, providing further logic compaction. Also, 24 of the flip-flops may be bypassed to provide internal combinatorial feedback to the logic array.

Product terms provide individual clocks and asynchronous resets for each flip-flop. The flip-flops may also be individually configured to have direct input pin clocking. Each output has its own enable product term. Eight synchronous preset product terms serve local groups of either four or eight flip-flops. Register preload functions are provided to simplify testing. All registers automatically reset upon power up.

The ATV2500BQL is the low power compatible device with speeds as fast as 20 ns. The ATV2500BQL consumes only 2 mA at standby, which provides the optimum low power PLD solution with full CMOS output levels. The ATV2500BQL significantly reduces total system power, allowing battery-powered operation.

D.C. and A.C. Operating Conditions

	Commercial	Industrial	Military
Operating Temperature (Case)	0°C - 70°C	-40°C - 85°C	-55°C - 125°C
VCC Power Supply	5 V ± 5%	5 V ± 10%	5 V ± 10%

1

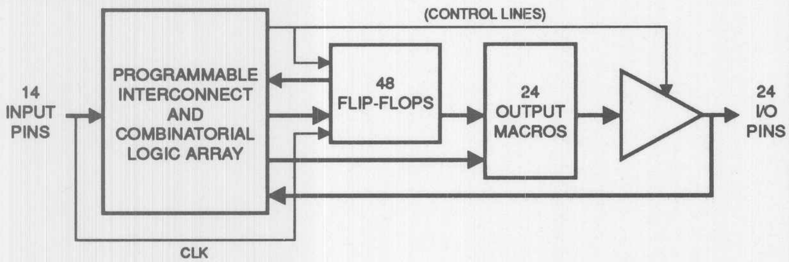
High Speed
High Density
UV Erasable
Programmable
Logic Device

Advanced Information

Features

- **3-Volt Operation**
- **Wide V_{CC} Range**
  - **V_{CC} = 3.0 V to 5.25 V (Commercial)**
  - **V_{CC} = 3.0 V to 5.5 V (Industrial)**
- **High Performance, High Density Programmable Logic Device**
  - **Maximum 15 ns Pin-to-Pin Delay**
  - **Fully Connected Logic Array With 416 Product Terms**
- **Flexible Output Macrocell**
  - **48 Flip-Flops - Two per Macrocell**
  - **72 Sum Terms**
  - **All Flip-Flops, I/O Pins Feed In Independently**
  - **Achieves Over 80% Gate Utilization**
- **Enhanced Macrocell Configuration Selections**
  - **D- or T-Type Flip-Flops**
  - **Product Term or Direct Input Pin Clocking**
  - **Registered or Combinatorial Internal Feedback**
- **Low Power ATLV2500BL - 7.2 mW Stand-By (Typical) at 3.6 V**
- **Backward Compatible With ATV2500H/L Software**
- **Proven and Reliable High Speed UV EPROM Process**
- **Reprogrammable - Tested 100% for Programmability**
- **44-Lead Surface Mount Packages**

Block Diagram



Description

The ATLV2500Bs are the highest density PLDs available in a 44-pin package. With their fully connected logic array and flexible macrocell structure, high gate utilization is easily obtainable.

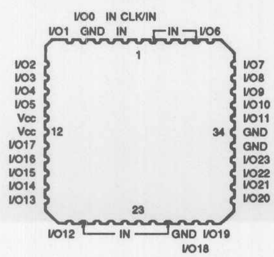
The ATLV2500Bs are organized around a *single universal input bus*. All pin and feedback terms are always available to every macrocell. Each of the 38 logic pins are array inputs, as are the outputs of each flip-flop.

(continued)

Pin Configurations

Pin Name	Function
IN	Logic Inputs
CLK/IN	Pin Clock and Input
I/O	Bidirectional Buffers
GND	Ground
Vcc	+5 V Supply

SMD



Description (Continued)

In the ATLV2500Bs, four product terms are input to each sum term. Furthermore, each macrocell's three sum terms can be combined to provide up to 12 product terms per sum term with *no performance penalty*. Each flip-flop is individually selectable to be either D- or T-type, providing further logic compaction. Also, 24 of the flip-flops may be bypassed to provide internal combinatorial feedback to the logic array.

Product terms provide individual clocks and asynchronous resets for each flip-flop. The flip-flops may also be individually configured to have direct input pin clocking. Each output has its own enable product term. Eight synchronous preset product terms serve local groups of either four or eight flip-flops. Register preload functions are provided to simplify testing. All registers automatically reset upon power up.

The ATLV2500BL is the low voltage compatible device with speeds as fast as 20 ns. The ATLV2500BL consumes only 2 mA at standby, which provides the optimum low power PLD solution with full CMOS output levels. The ATLV2500BL significantly reduces total system power, allowing battery-powered operation.

D.C. and A.C. Operating Conditions

	Commercial -15, -20	Industrial -15, -20
Operating Temperature (Case)	$0^{o}C$ - $70^{o}C$	-$40^{o}C$ - $85^{o}C$
V_{CC} Power Supply	3.0 V to 5.25 V	3.0 V to 5.5 V

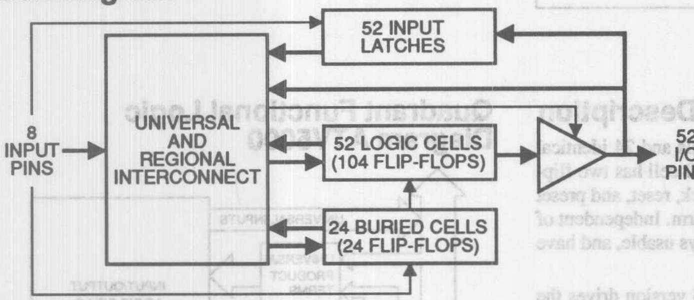
1

High Density UV Erasable Programmable Logic Device

Features

- Advanced Programmable Logic Device - High Gate Utilization
- Flexible Interconnect Architecture - Universal Routing
- Flexible Logic Cells - 128 Flip-Flops and 52 Latches
- Multiple Flip-Flop Types - Synchronous or Asynchronous Registers
- High Speed - 50 MHz Operation
- Complete Third Party Software Support
  No Placement, Routing or Layout Software Required
- Proven and Reliable High Speed CMOS EPROM Process
  2000 V ESD Protection
  200 mA Latchup Immunity
- Reprogrammable - Tested 100% for Programmability
- Commercial, Industrial and Military Temperature Grades

Block Diagram



Description

The Atmel V5000 is an easy to use, high density programmable logic device. Its simple, regular architecture translates into increased utilization and high performance.

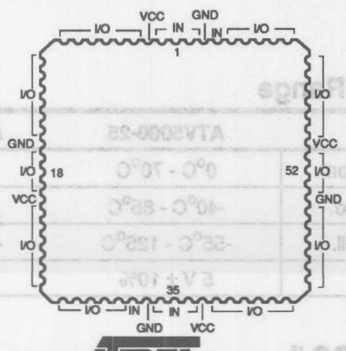
The ATV5000 has one programmable combinatorial logic array. This guarantees easy interconnection of and uniform performance from all nodes. "Sum terms", which are easy to use groupings of AND-OR gates, provide combinatorial logic blocks. Sum terms can be wire-OR'd together to integrate larger logic blocks. To expand the levels of logic, buried sum terms feed back into the logic array. The 52 I/O pins can each be driven by a register or a sum term. Each I/O pin has an individually enabled input latch.

All 128 registers are configurable as D- or T-types without using extra logic gates. Individual sum terms, asynchronous presets, resets and clocks give each flip-flop added flexibility. A direct "clock from pin" option guarantees synchronization and fast clock to output performance.

Standard, off-the-shelf third-party software tools and programmers support the ATV5000. This minimizes start-up investment and improves product support.

Chip Carrier Pin Configuration

Pin Name	Function
IN	Logic and Clock Inputs
Pins 2,32,36,66	Input/Register Clocks 1-4
Pins 1,34,35,68	Input/Latch Clocks 1-4
I/O	Bidirectional Buffers
VCC	+5 V Supply



Absolute Maximum Ratings*

Temperature Under Bias	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-2.0 V to +7.0 V[1]
Voltage on Input Pins with Respect to Ground During Programming	-2.0 V to +14.0 V[1]
Programming Voltage with Respect to Ground	-2.0 V to +14.0 V[1]
Integrated UV Erase Dose	7258 W·sec/cm^2

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Note:
1. Minimum voltage is -0.6 V dc which may undershoot to -2.0 V for pulses of less than 20 ns. Maximum output pin voltage is V_{CC}+0.75 V dc which may overshoot to +7.0 V for pulses of less than 20 ns.

Functional Logic Diagram Description

There are 52 identical input/ouput logic cells and 24 identical buried logic cells in the ATV5000. Each I/O cell has two flip-flops, up to three sum terms, individual clock, reset, and preset terms per flip-flop, and one output enable term. Independent of output configuration, all flip-flops are always usable, and have at least four product term inputs each.

Each I/O pin (52 total) signal or its latched version drives the logic array. There is one latch clock per quadrant.

The ATV5000 has four identical quadrants (see Figure 2). The universal bus routes true and false signals from each of the 52 I/O pins to all four quadrants. Regional buses route each quadrant's flip-flop Q and $\overline{Q}$ locally. The eight input-only pins are available in all four regional buses.

Each logic cell has a number of "regional" and "universal" product terms (see Figure 1). The I/O logic cells contain three sum terms, two flip-flops, and an I/O buffer.

The buried logic cells each contain one flip-flop. In addition, in each buried logic cell the sum term can drive the regional bus. This allows for logic expansion.

Serial register preload and observability simplify testing. All registers automatically clear at power up.

Quadrant Functional Logic Diagram ATV5000

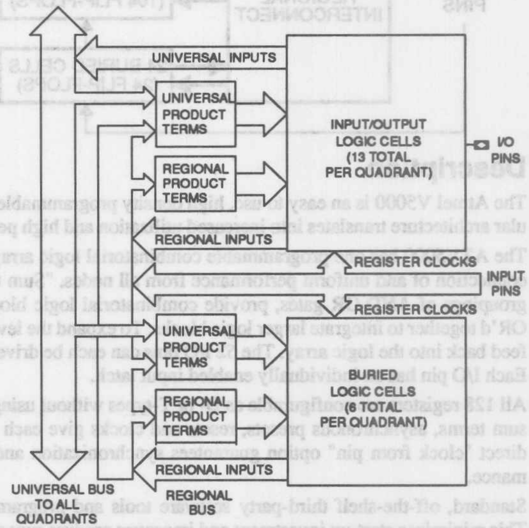


Figure 1

D.C. and A.C. Operating Range

		ATV5000-25	ATV5000/L-30	ATV5000/L-35
Operating Temperature (Case)	Com.	0°C - 70°C	0°C - 70°C	0°C - 70°C
	Ind.	-40°C - 85°C	-40°C - 85°C	-40°C - 85°C
	Mil.	-55°C - 125°C	-55°C - 125°C	-55°C - 125°C
V_{CC} Power Supply		5 V ± 10%	5 V ± 10%	5 V ± 10%

1

ATV5000 Block Diagram

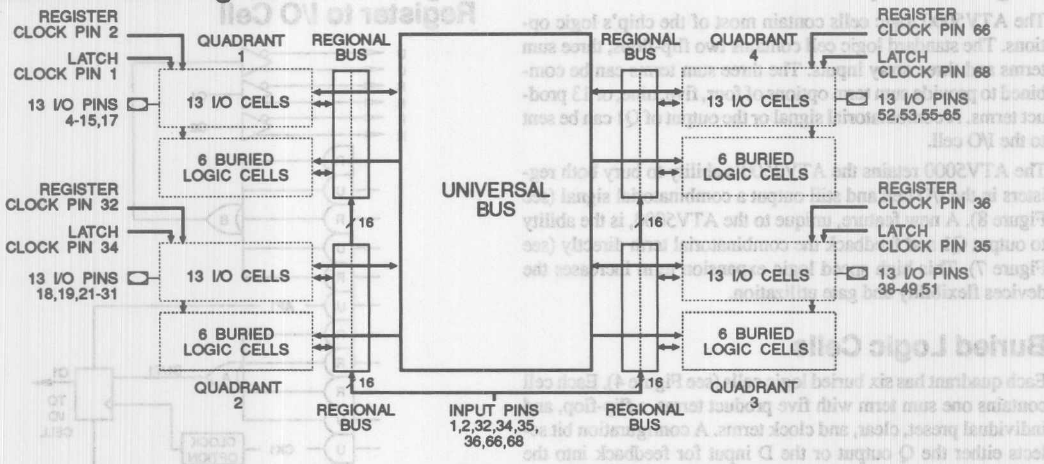


Figure 2

Quadrant Logic Diagram and Description

The ATV5000 has: four identical quadrants, 52 identical input/output logic cells, and 24 identical buried logic cells. The universal bus routes true and false signals from each of the 52 I/O pins to all four quadrants. Regional buses route each quadrant's flip-flop Q and $\overline{Q}$ locally. The eight input-only pins are available in every regional bus.

Each logic cell has a number of "regional" and "universal" product terms (see Figure 3). The I/O logic cells (Figures 7, 8, 9) contain three sum terms, two flip-flops, and an I/O buffer. Sum term B has five product terms - two universal and three regional. Sum terms A and C each have four product terms - one universal and three regional. Flip-flop Q1 has global asynchronous preset, reset, and clock product terms. Flip-flop Q2 has universal asynchronous reset and clock terms and a regional asynchronous preset term. There is one universal product term for the I/O pin output enable.

The buried logic cells (Figure 4) each contain one flip-flop. The sum term has one universal product term and four regional product terms for a total of five. The flip-flop has universal asynchronous preset, reset, and clock terms. In addition, in each buried logic cell the sum term can be fed back into the regional bus instead of the flip-flop. This allows for logic expansion.

Regional product terms have as inputs all quadrant flip-flop outputs (or buried flip-flop inputs) and the eight dedicated input pins. Universal product terms have the same inputs plus the 52 I/O pins and their complements.

Quadrant Clock Pin Assignments

Quadrant Number	Register Clock Pin	Latch Clock Pin
1	2	1
2	32	34
3	36	35
4	66	68

Quadrant Structure

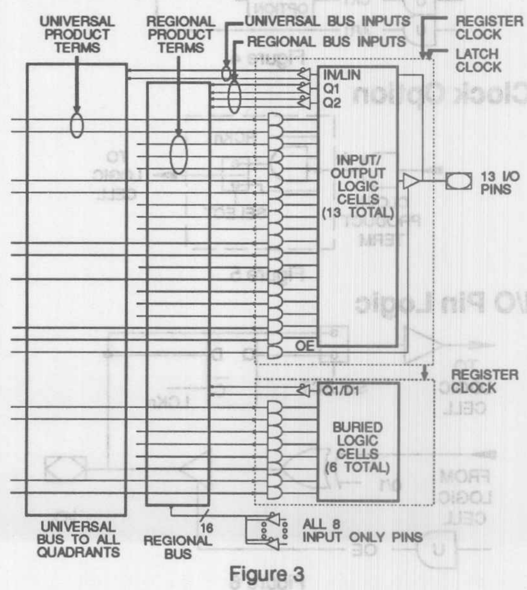


Figure 3

Logic Cell Options

The ATV5000 logic cells contain most of the chip's logic options. The standard logic cell contains two flip-flops, three sum terms and three array inputs. The three sum terms can be combined to provide sum term options of four, five, nine, or 13 product terms. A combinatorial signal or the output of Q1 can be sent to the I/O cell.

The ATV5000 retains the ATV2500's ability to bury both registers in the I/O cell and still output a combinatorial signal (see Figure 8). A new feature, unique to the ATV5000, is the ability to output Q1 and feedback the combinatorial term directly (see Figure 7). This high speed logic expansion term increases the devices flexibility and gate utilization.

Buried Logic Cells

Each quadrant has six buried logic cells (see Figure 4). Each cell contains one sum term with five product terms, a flip-flop, and individual preset, clear, and clock terms. A configuration bit selects either the Q output or the D input for feedback into the regional bus.

Buried Logic Cells

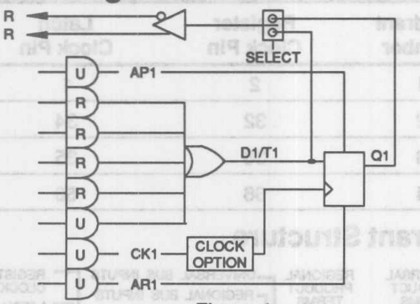


Figure 4

Clock Option

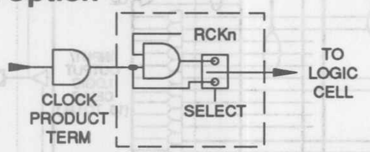


Figure 5

I/O Pin Logic

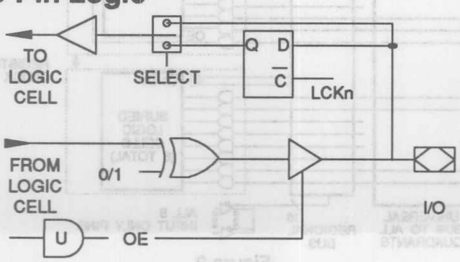


Figure 6

Logic Cell with Buried Sum Term and Register to I/O Cell

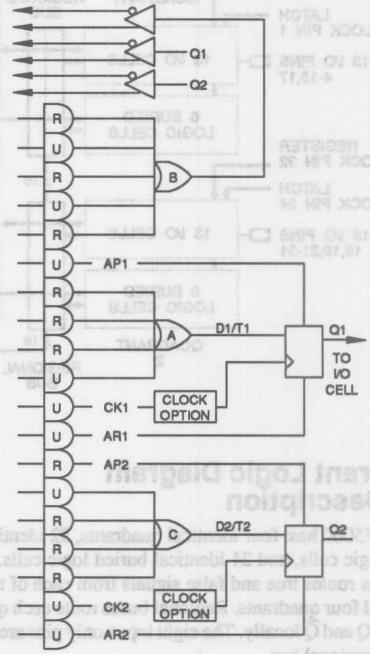


Figure 7

Flip-Flop Clock Options

Each register may be connected to its regional clock to provide fast clock-to-output timing (see Figure 5). In this "synchronous" mode, the clock is one of four input pins, a unique clock pin for each chip quadrant. One product term defines each flip-flop's clock in the "asynchronous" mode.

In the "synchronous" mode, the regional clock is ANDed with the product term. This provides the fast timing of a synchronous clock with the local control of the product term.

I/O Pin Latches

Each I/O pin of the ATV5000 has an input latch which can be individually enabled or disabled (see Figure 6). Each chip quadrant has a unique latch clock. When the latch is inactive, pin input flows directly into the array. When activated, the latch is flow-through when the clock signal is low, and data is captured on the clock's rising edge.

Flip-Flop Types

Each flip-flop in the ATV5000 may be configured as either a T- or D-type flip-flop. A T-type flip-flop can also easily be configured into a JK or SR flip-flop.

Logic Cell, Two Buried Registers, Combinatorial to I/O Cell

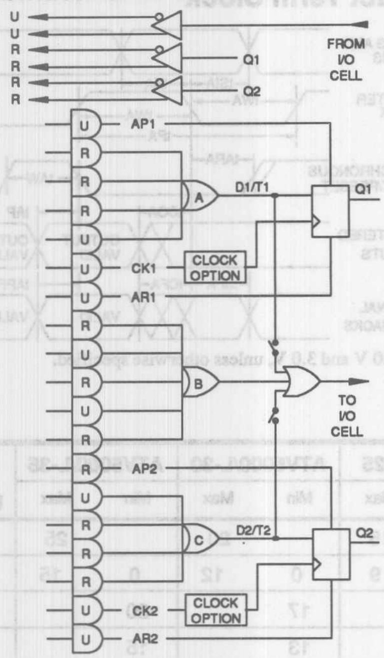


Figure 8

Logic Cell with Combinable Sum Terms, Register to I/O Cell

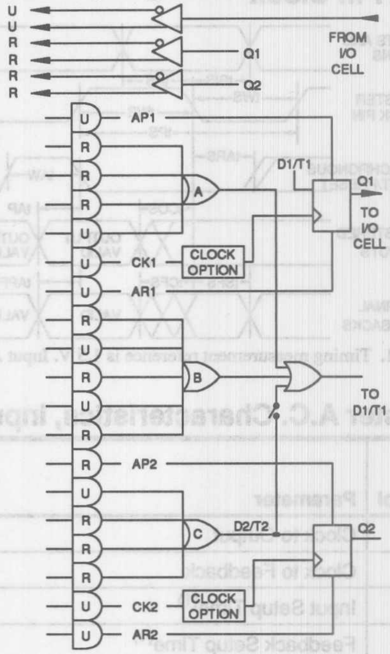


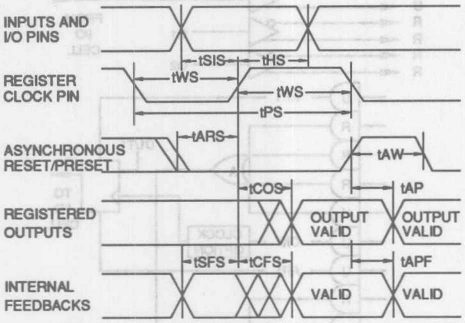
Figure 9

D.C. Characteristics

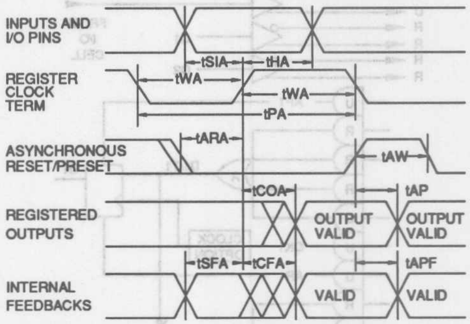
Symbol	Parameter	Condition		Min	Typ	Max	Units
I_{LI}	Input Load Current	V_{IN} = -0.1 V to V_{CC}+1 V				10	µA
I_{LO}	Output Leakage Current	V_{OUT} = -0.1 V to V_{CC}+0.1 V				10	µA
I_{CC}	Power Supply Current ATV5000	V_{CC} = MAX, V_{IN} = GND or V_{CC} Outputs Open	Com.		200	350	mA
			Ind.,Mil.		200	400	mA
I_{CC}	Power Supply Current ATV5000L	V_{CC} = MAX, V_{IN} = GND or V_{CC} Outputs Open	Com.		32	40	mA
			Ind.,Mil.		32	50	mA
I_{CC2}	Clocked Power Supply Current, ATV5000L Only	f = 1 MHz, V_{CC} = MAX Outputs Open	Com.		30	50	mA
			Ind.,Mil.		30	60	mA
I_{OS} (1)	Output Short Circuit Current	V_{OUT} = 0.5 V				-90	mA
V_{IL}	Input Low Voltage			-0.6		0.8	V
V_{IH}	Input High Voltage			2.0		V_{CC}+0.75	V
V_{OL}	Output Low Voltage	V_{IN} = V_{IH} or V_{IL}, I_{OL} = 8 mA Com,Ind; 6 mA Mil.				0.5	V
V_{OH}	Output High Voltage	I_{OH} = -100 µA		V_{CC}-0.3			V
		I_{OH} = -4.0 mA		2.4			V

Note: 1. Not more than one output at a time should be shorted. Duration of short circuit test should not exceed 30 seconds.

A.C. Waveforms[1] Input Pin Clock



A.C. Waveforms[1] Product Term Clock



Notes: 1. Timing measurement reference is 1.5 V. Input AC driving levels are 0.0 V and 3.0 V, unless otherwise specified.

Register A.C. Characteristics, Input Pin Clock

Symbol	Parameter	ATV5000-25		ATV5000/L-30		ATV5000/L-35		Units
		Min	Max	Min	Max	Min	Max	
t_{COS}	Clock to Output		15		20		25	ns
t_{CFS}	Clock to Feedback	0	9	0	12	0	15	ns
t_{SIS}	Input Setup Time[1]	16		17		20		ns
t_{SFS}	Feedback Setup Time[1]	11		13		15		ns
t_{HS}	Hold Time	0		0		0		ns
t_{WS}	Clock Width	10		12		15		ns
t_{PS}	Clock Period	20		25		30		ns
F_{MAXS}	Maximum Frequency ($1/t_{PS}$)		50		40		33	MHz
t_{ARS}	Asynchronous Reset/Preset Recovery Time	20		25		30		ns

Note: 1. Add 3 ns for Universal Product Terms.

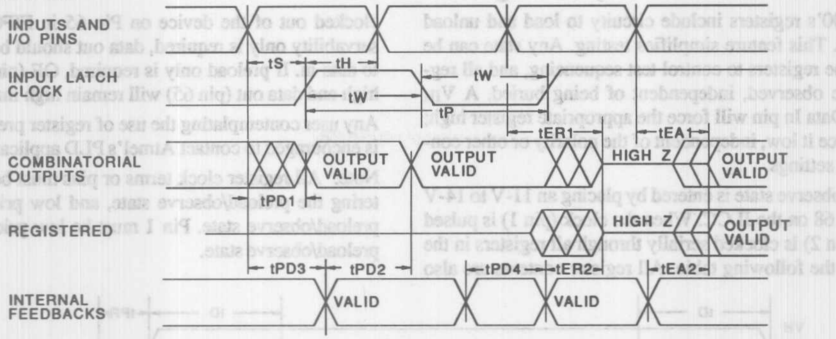
Register A.C. Characteristics, Product Term Clock

Symbol	Parameter	ATV5000-25		ATV5000/L-30		ATV5000/L-35		Units
		Min	Max	Min	Max	Min	Max	
t_{COA}	Clock to Output		25		30		35	ns
t_{CFA}	Clock to Feedback	7	20	10	25	12	27	ns
t_{SIA}	Input Setup Time[1]	10		12		15		ns
t_{SFA}	Feedback Setup Time[1]	5		8		13		ns
t_{HA}	Hold Time	8		10		12		ns
t_{WA}	Clock Width	12		15		15		ns
t_{PA}	Clock Period	25		33		40		ns
F_{MAXA}	Maximum Frequency ($1/t_{PA}$)		40		30		25	MHz
t_{ARA}	Asynchronous Reset/Preset Recovery Time	15		20		25		ns

Note: 1. Add 3 ns for Universal Product Terms.

1

A.C. Waveforms [1]



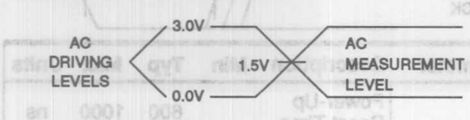
Notes: 1. Timing measurement reference is 1.5 V. Input AC driving levels are 0.0 V and 3.0 V, unless otherwise specified.

A.C. Characteristics

Symbol	Parameter	ATV5000-25 Min	ATV5000-25 Max	ATV5000/L-30 Min	ATV5000/L-30 Max	ATV5000/L-35 Min	ATV5000/L-35 Max	Units
t_{PD1}	Input to Non-Registered Output[1]		25		30		35	ns
t_{PD2}	Feedback to Non-Registered Output[1]		20		25		30	ns
t_{PD3}	Input to Non-Registered Feedback[1]		20		25		30	ns
t_{PD4}	Feedback to Non-Registered Feedback[1]		15		18		22	ns
t_{EA1}	Input to Output Enable		30		35		40	ns
t_{ER1}	Input to Output Disable		30		35		40	ns
t_{EA2}	Feedback to Output Enable		25		30		35	ns
t_{ER2}	Feedback to Output Disable		25		30		35	ns
t_S	Input Latch Setup Time	5		6		7		ns
t_H	Input Latch Hold Time	5		5		5		ns
t_W	Clock Width	10		12		12		ns
t_P	Clock Period	20		25		30		ns
F_{MAX}	Maximum Frequency ($1/t_P$)		50		40		33	MHz
t_{AW}	Asynchronous Reset/Preset Width	15		20		20		ns
t_{AP}	Asynchronous Reset/ Preset to Registered Output		30		35		40	ns
t_{APF}	Asynchronous Reset/ Preset to Registered Feedback		25		30		35	ns

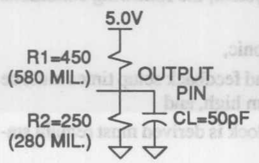
Note: 1. Add 3 ns for Universal Product Terms.

Input Test Waveforms and Measurement Levels



t_R, t_F < 5 ns (10% to 90%)

Output Test Load



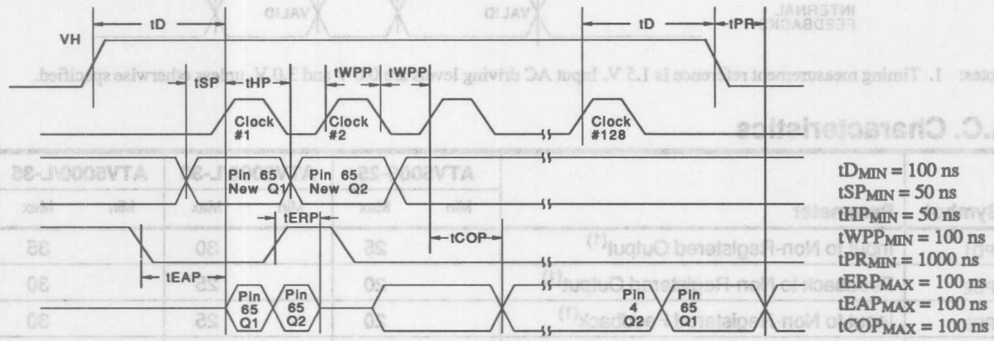
Preload and Observability of Registers

The ATV5000's registers include circuity to load and unload them serially. This feature simplifies testing. Any state can be forced into the registers to control test sequencing, and all registers may be observed, independent of being buried. A V_{IH} level on the Data In pin will force the appropriate register high; a V_{IL} will force it low, independent of the polarity or other configuration bit settings.

The preload/observe state is entered by placing an 11-V to 14-V signal on pin 68 on the JLCC. When the clock (pin 1) is pulsed high, data (pin 2) is clocked serially through all registers in the device, as in the following table. All register contents are also clocked out of the device on Pin 65 in FIFO fashion. If observability only is required, data out should be connected back to data in. If preload only is required, OE (pin 66) can be held high and data out (pin 65) will remain high impedance.

Any user contemplating the use of register preload/obervability is encouraged to contact Atmel's PLD applications department.

Note: All register clock terms or pins must be low prior to entering the preload/observe state, and low prior to leaving the preload/observe state. Pin 1 must be low prior to entering the preload/observe state.



Preload / Observe Register Scan Order

Quadrant	Pin	
Quadrant 1	Pin	4 5 6 ••• 15 17
	D_{IN}	→ Q2 → Q1 → B23 → Q2 → Q1 → Q2 → Q1 ••• B18 → Q2 → Q1 → Q2 → Q1 → (Quadrant 2)
Quadrant 2	Pin	18 19 21 22 ••• 31
(Quadrant 1)→		Q2 → Q1 → Q2 → Q1 → B17 → Q2 → Q1 → Q2 → Q1 ••• B12 → Q2 → Q1 → (Quadrant 3)
Quadrant 3	Pin	38 39 40 ••• 49 51
(Quadrant 2)→		Q2 → Q1 → B11 → Q2 → Q1 → Q2 → Q1 ••• B6 → Q2 → Q1 → Q2 → Q1 → (Quadrant 4)
Quadrant 4	Pin	52 53 55 56 ••• 65
(Quadrant 3)→		Q2 → Q1 → Q2 → Q1 → B5 → Q2 → Q1 → Q2 → Q1 ••• B0 → Q2 → Q1 → D_{OUT}

Power Up Reset

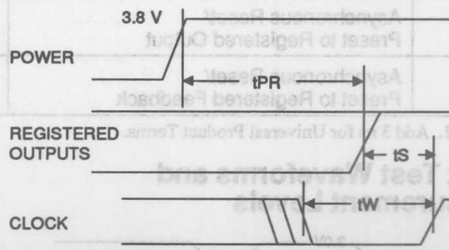
The registers in the ATV5000 are designed to reset during power up. At a point delayed slightly from V_{CC} crossing 3.8 V, all registers will be reset to the low state. The output state will depend on the polarity of the output buffer.

This feature is critical for state machine initialization. However, due to the asynchronous nature of reset and the uncertainty of how V_{CC} actually rises in the system, the following conditions are required:

1) The V_{CC} rise must be monotonic,

2) After reset occurs, all input and feedback setup times must be met before driving the clock term high, and

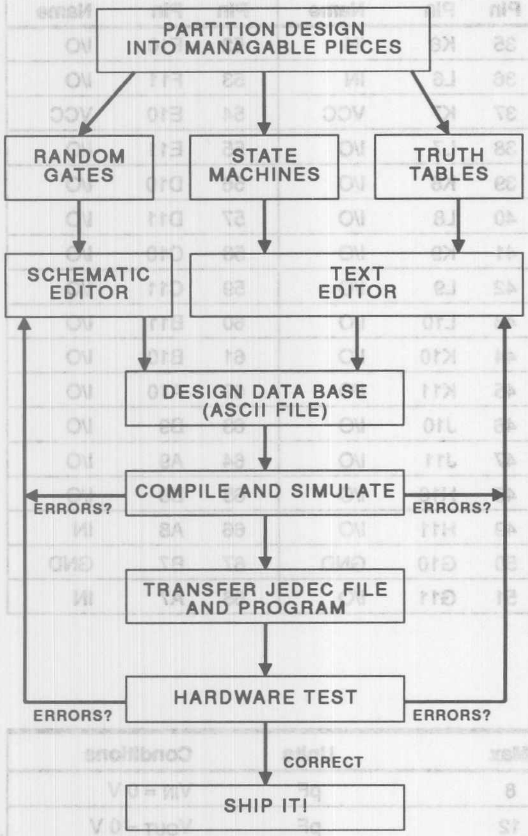
3) The signals from which the clock is derived must remain stable during t_{PR}.



Parameter	Description	Min	Typ	Max	Units
t_{PR}	Power-Up Reset Time		600	1000	ns

1

Design Flow Diagram



Using The ATV5000

The ATV5000's simple, regular architecture means that only simple logic compilers are required to configure the device. No layout or route and place are required. These software tools are readily available from companies such as Data I/O Corporation (Abel™), Logical Devices (Cupl™), and ISDATA (LOGiC™).

The first step in designing a device as complex as the ATV5000 is to partition your design into manageable blocks. These blocks are then allocated proportionally to each of the four quadrants of the ATV5000. Random gates can be described either with boolean equations (a behavioral description) or with a schematic editor. Truth table logic and state machines are best described behaviorially and entered with a text editor. The design is then combined into one ASCII file, which is then submitted to the logic compiler. Compilation, logic reduction, simulation, JEDEC file creation and documentation are then completed by all of the popular compilers.

Assignment of signals to pins or buried nodes as well as selecting the various options of the ATV5000 (such as register clocks and input latches) can be done manually in the design data base file, or an automatic fitter may be used.

A logic fitter assigns pins and nodes to make best use of the features in the ATV5000, and frees the designer from being required to learn all of the features of a complex device such as the ATV5000. For further information on fitters for the ATV5000, contact Atmel's PLD applications department.

After correcting any syntax and logic errors discovered by the compiler, the JEDEC file is ready to download to an PLD programmer. These are available from a number of manufacturers. Programmed devices are usually first tested in the programmer with your supplied test vectors. The next step is check out your "custom chip" in the target system. When this hardware debug step is complete, your system is ready to go— all in a matter of hours.

Abel™, Cupl™ and LOGiC™ may be trademarks of others.

Operating Modes

	68-Lead LCC Pin								
Mode	1	2	36	34	68	66	VCC (3,20,37,54)	I/O's	I/O Pin 65
"PLD"	X (1)	X	X	X	X	X	5V	I/O	I/O
Program	V_{PP}	V_{IL}	V_{IL}	V_{IH}	V_{IH}	X	6V	ADD/D_{IN}	ADD
PGM Verify	V_{PP}	V_{IH}	V_{IH}	V_{IL}	V_{IL}	X	6V	ADD/D_{OUT}	ADD
PGM Inhibit	V_{PP}	V_{IH}	V_{IH}	V_{IH}	V_{IH}	X	6V	High Z	High Z
Preload/Observe	⎍	D_{IN}	X	X	V_H (2)	$\overline{OE}$	5V	High Z	D_{OUT}

Notes: 1. X can be V_{IL} or V_{IH}.
2. V_H = 11.0 V to 14.0 V

ATV5000 PLCC/PGA Pin Assignments

PLCC Pin	PGA Pin	Name	PLCC Pin	PGA Pin	Name	PLCC Pin	PGA Pin	Name	PLCC Pin	PGA Pin	Name
1	B6	IN	18	F2	I/O	35	K6	IN	52	F10	I/O
2	A6	IN	19	F1	I/O	36	L6	IN	53	F11	I/O
3	B5	VCC	20	G2	VCC	37	K7	VCC	54	E10	VCC
4	A5	I/O	21	G1	I/O	38	L7	I/O	55	E11	I/O
5	B4	I/O	22	H2	I/O	39	K8	I/O	56	D10	I/O
6	A4	I/O	23	H1	I/O	40	L8	I/O	57	D11	I/O
7	B3	I/O	24	J2	I/O	41	K9	I/O	58	C10	I/O
8	A3	I/O	25	J1	I/O	42	L9	I/O	59	C11	I/O
9	A2	I/O	26	K1	I/O	43	L10	I/O	60	B11	I/O
10	B2	I/O	27	K2	I/O	44	K10	I/O	61	B10	I/O
11	B1	I/O	28	L2	I/O	45	K11	I/O	62	A10	I/O
12	C2	I/O	29	K3	I/O	46	J10	I/O	63	B9	I/O
13	C1	I/O	30	L3	I/O	47	J11	I/O	64	A9	I/O
14	D2	I/O	31	K4	I/O	48	H10	I/O	65	B8	I/O
15	D1	I/O	32	L4	IN	49	H11	I/O	66	A8	IN
16	E2	GND	33	K5	GND	50	G10	GND	67	B7	GND
17	E1	I/O	34	L5	IN	51	G11	I/O	68	A7	IN

Pin Capacitance (f = 1 MHz, T = 25˚C) (1)

	Typ	Max	Units	Conditions
C_{IN}	6	8	pF	$V_{IN} = 0$ V
C_{OUT}	8	12	pF	$V_{OUT} = 0$ V

Note: 1. Typical values for nominal supply voltage. This parameter is only sampled and is not 100% tested.

Security Fuse Usage

A single fuse is provided to prevent unauthorized copying of the ATV5000 fuse patterns. Once programmed, all outputs appear programmed during verify. The security fuse should be programmed last (after verifying all other programmed bits), as its effect is immediate.

The security fuse also inhibits preload and observability.

Erasure Characteristics

The entire memory array of an ATV5000 is erased after exposure to ultraviolet light at a wavelength of 2537 Å. Complete erasure is assured after a minimum of 20 minutes exposure using 12,000 μW/cm^2 intensity lamps spaced one inch away from the chip. Minimum erase time for lamps at other intensity ratings can be calculated from the minimum integrated erasure dose of 15 W·sec/cm^2. To prevent unintentional erasure, an opaque label is recommended to cover the clear window on any UV erasable PLD which will be subjected to continuous fluorescent indoor lighting or sunlight.

Ordering Information

t_{PD} (ns)	t_{COS} (ns)	f_{MAX} (MHz)	Ordering Code	Package	Operation Range
25	15	50	ATV5000-25JC ATV5000-25KC ATV5000-25UC	68J 68KW 68UW	Commercial (0°C to 70°C)
			ATV5000-25KI ATV5000-25UI	68KW 68UW	Industrial (-40°C to 85°C)
			ATV5000-25KM ATV5000-25UM	68KW 68UW	Military (-55°C to 125°C)
30	20	40	ATV5000-30JC ATV5000-30KC ATV5000-30UC	68J 68KW 68UW	Commercial (0°C to 70°C)
			ATV5000-30KI ATV5000-30UI	68KW 68UW	Industrial (-40°C to 85°C)
			ATV5000-30KM ATV5000-30UM	68KW 68UW	Military (-55°C to 125°C)
			ATV5000-30KM/883 ATV5000-30UM/883	68KW 68UW	Military/883C Class B, Fully Compliant (-55°C to 125°C)
35	25	33	ATV5000-35JC ATV5000-35KC ATV5000-35UC	68J 68KW 68UW	Commercial (0°C to 70°C)
			ATV5000-35KI ATV5000-35UI	68KW 68UW	Industrial (-40°C to 85°C)
			ATV5000-35KM ATV5000-35UM	68KW 68UW	Military (-55°C to 125°C)
			ATV5000-35KM/883 ATV5000-35UM/883	68KW 68UW	Military/883C Class B, Fully Compliant (-55°C to 125°C)

Ordering Information

t_{PD} (ns)	t_{COS} (ns)	f_{MAX} (MHz)	Ordering Code	Package	Operation Range
30	20	40	ATV5000L-30JC ATV5000L-30KC ATV5000L-30UC	68J 68KW 68UW	Commercial (0°C to 70°C)
35	25	33	ATV5000L-35JC ATV5000L-35KC ATV5000L-35UC	68J 68KW 68UW	Commercial (0°C to 70°C)
			ATV5000L-35KI ATV5000L-35UI	68KW 68UW	Industrial (-40°C to 85°C)
			ATV5000L-35KM ATV5000L-35UM	68KW 68UW	Military (-55°C to 125°C)
			ATV5000L-35KM/883 ATV5000L-35UM/883	68KW 68UW	Military/883C Class B, Fully Compliant (-55°C to 125°C)

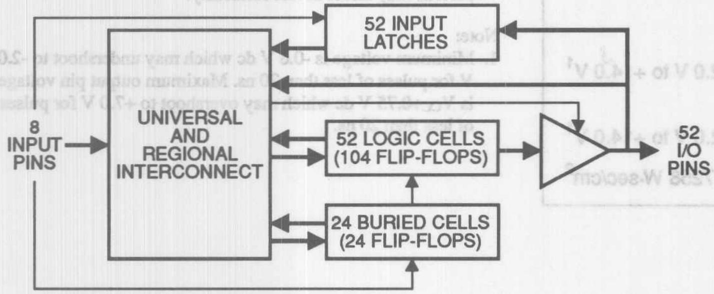
Package Type	
68J	68 Lead, Plastic J-Leaded Chip Carrier OTP (PLCC)
68KW	68 Lead, Windowed, Ceramic J-Leaded Chip Carrier (JLCC)
68UW	68 Pin, Windowed, Ceramic Pin Grid Array (PGA)

High Density UV Erasable Programmable Logic Device

Features

- Advanced Programmable Logic Device - High Gate Utilization
- Pin-Compatible with Atmel's ATV5000
- Flexible Interconnect Architecture - Similar to ATV5000 with More Emphasis on Universal Routing
- Flexible Logic Cells - 128 Flip-Flops and 52 Latches
- Multiple Flip-Flop Types - Synchronous or Asynchronous Registers
- High Speed - 50 MHz Operation
- Complete Third Party Software Support
  No Placement, Routing or Layout Software Required
- Proven and Reliable High Speed CMOS EPROM Process
  2000 V ESD Protection
  200 mA Latchup Immunity
- Reprogrammable - Tested 100% for Programmability
- Commercial, Industrial and Military Temperature Grades

Block Diagram



Description

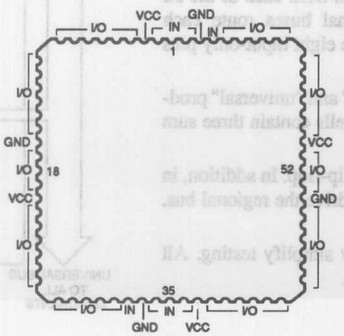
The Atmel V5100 is an easy to use, high density programmable logic device. Its simple, regular architecture translates into increased utilization and high performance. Additional universal routing simplifies design fitting.

The ATV5100 has one programmable combinatorial logic array. This guarantees easy interconnection of and uniform performance from all nodes. "Sum terms", which are easy to use groupings of AND-OR gates, provide combinatorial logic blocks. Sum terms can be wire-OR'd together to integrate larger logic blocks. To expand the levels of logic, buried sum terms

continued on next page

Chip Carrier Pin Configuration

Pin Name	Function
IN	Logic and Clock Inputs
Pins 2,32,36,66	Input/Register Clocks 1-4
Pins 1,34,35,68	Input/Latch Clocks 1-4
I/O	Bidirectional Buffers
VCC	+5 V Supply



Description (Continued)

feed back into the logic array. The 52 I/O pins can each be driven by a register or a sum term. Each I/O pin has an individually enabled input latch.

All 128 registers are configurable as D- or T-types without using extra logic gates. Individual sum terms, asynchronous presets, resets and clocks give each flip-flop added flexibility. A direct "clock from pin" option guarantees synchronization and fast clock to output performance.

Standard, off-the-shelf third-party software tools and programmers support the ATV5100. This minimizes start-up investment and improves product support.

Absolute Maximum Ratings*

Temperature Under Bias	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-2.0 V to +7.0 V[1]
Voltage on Input Pins with Respect to Ground During Programming	-2.0 V to +14.0 V[1]
Programming Voltage with Respect to Ground	-2.0 V to +14.0 V[1]
Integrated UV Erase Dose	7258 W·sec/cm^2

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Note:
1. Minimum voltage is -0.6 V dc which may undershoot to -2.0 V for pulses of less than 20 ns. Maximum output pin voltage is V_{CC}+0.75 V dc which may overshoot to +7.0 V for pulses of less than 20 ns.

Functional Logic Diagram Description

There are 52 identical input/ouput logic cells and 24 identical buried logic cells in the ATV5100. Each I/O cell has two flip-flops, up to three sum terms, individual clock, reset, and preset terms per flip-flop, and one output enable term. Independent of output configuration, all flip-flops are always usable, and have at least four product term inputs each.

Each I/O pin (52 total) signal or its latched version drives the logic array. There is one latch clock per quadrant.

The ATV5100 has four identical quadrants (see Figure 2). The universal bus routes true and false signals from each of the 52 I/O pins to all four quadrants. Regional buses route each quadrant's flip-flop Q and $\overline{Q}$ locally. The eight input-only pins are available in all four regional buses.

Each logic cell has a number of "regional" and "universal" product terms (see Figure 1). The I/O logic cells contain three sum terms, two flip-flops, and an I/O buffer.

The buried logic cells each contain one flip-flop. In addition, in each buried logic cell the sum term can drive the regional bus. This allows for logic expansion.

Serial register preload and observability simplify testing. All registers automatically clear at power up.

Quadrant Functional Logic Diagram ATV5100

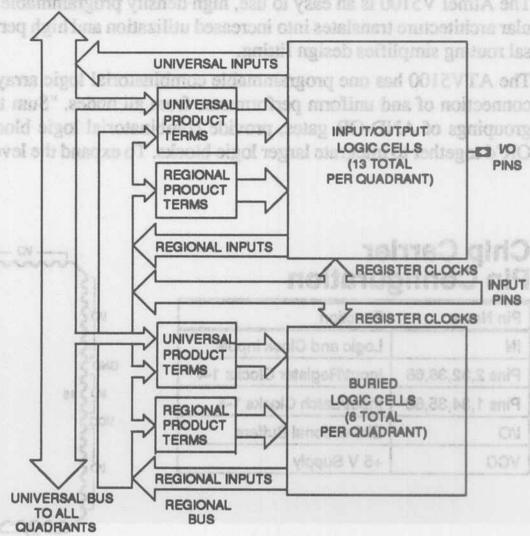


Figure 1

D.C. and A.C. Operating Range

		ATV5100-25	ATV5100/L-30	ATV5100/L-35
Operating Temperature (Case)	Com.	0°C - 70°C	0°C - 70°C	0°C - 70°C
	Ind.	-40°C - 85°C	-40°C - 85°C	-40°C - 85°C
	Mil.	-55°C - 125°C	-55°C - 125°C	-55°C - 125°C
V_{CC} Power Supply		5 V ± 10%	5 V ± 10%	5 V ± 10%

1

ATV5100 Block Diagram

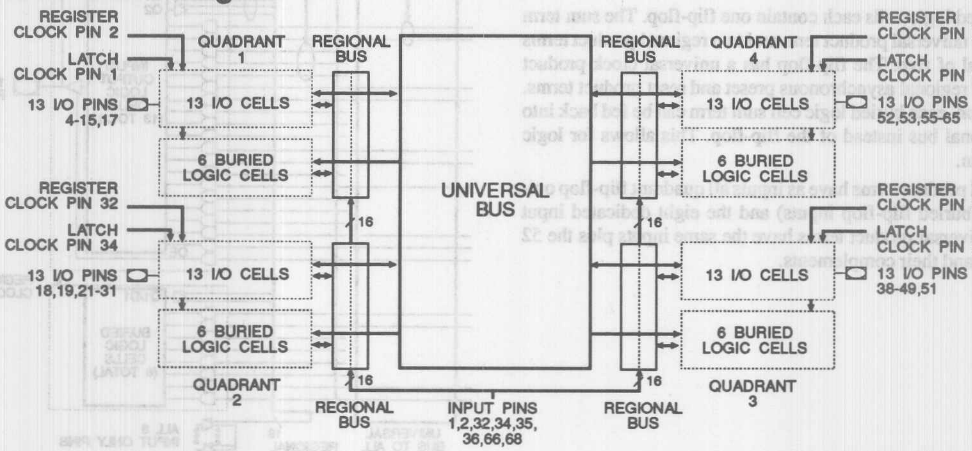


Figure 2

Quadrant Logic Diagram and Description

The ATV5100 has: four identical quadrants, 52 identical input/output logic cells, and 24 identical buried logic cells. The universal bus routes true and false signals from each of the 52 I/O pins to all four quadrants. Regional buses route each quadrant's flip-flop Q and $\overline{Q}$ locally. The eight input-only pins are available in every regional bus.

Each logic cell has a number of "regional" and "universal" product terms. The I/O logic cells contain three sum terms, two flip-flops, and an I/O buffer. Sum term B has five product terms - three universal and two regional. Sum term A has four product terms - two universal and two regional. Sum term C has four product terms - three universal and one regional (see next page). Flip-flops Q1 and Q2 have universal clock and regional preset and reset product terms. There is one regional product term for the I/O pin output enable.

The buried logic cells each contain one flip-flop. The sum term has three universal product term and two regional product terms for a total of five. The flip-flop has a universal clock product term and regional asynchronous preset and reset product terms. In addition, each buried logic cell sum term can be fed back into the regional bus instead of the flip-flop. This allows for logic expansion.

Regional product terms have as inputs all quadrant flip-flop outputs (or buried flip-flop inputs) and the eight dedicated input pins. Universal product terms have the same inputs plus the 52 I/O pins and their complements.

Quadrant Clock Pin Assignments

Quadrant Number	Register Clock Pin	Latch Clock Pin
1	2	1
2	32	34
3	36	35
4	66	68

Quadrant Logic Diagram

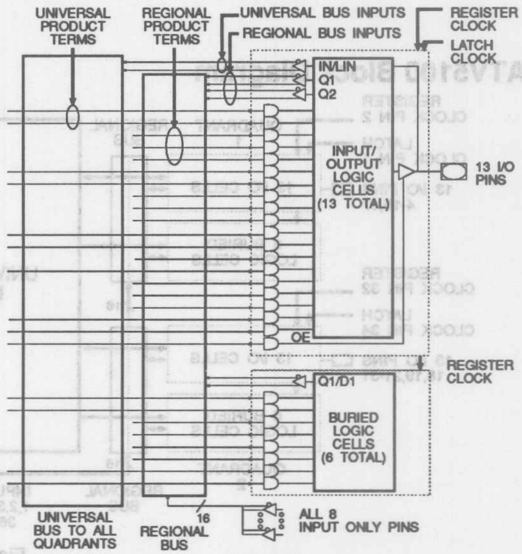


Figure 3

Logic Cell Options

The ATV5100 logic cells contain most of the chip's logic options. The standard logic cell contains two flip-flops, three sum terms and three array inputs. The three sum terms can be combined to provide sum term options of four, five, nine, or 13 product terms. A combinatorial signal or the output of Q1 can be sent to the I/O cell.

The ATV5100 retains the ATV2500's ability to bury both registers in the I/O cell and still output a combinatorial signal (see Figure 8). A new feature, unique to the ATV5000 and ATV5100, is the ability to output Q1 and feedback the combinatorial term directly (see Figure 7). This high speed logic expansion term increases the devices flexibility and gate utilization.

Buried Logic Cells

Each quadrant has six buried logic cells (see Figure 4). Each cell contains one sum term with five product terms, a flip-flop, and individual preset, clear, and clock terms. A configuration bit selects either the Q output or the D input for feedback into the regional bus.

Buried Logic Cells

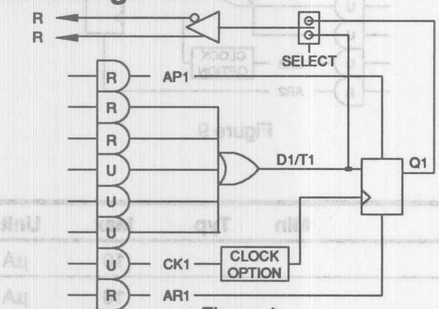


Figure 4

Clock Option

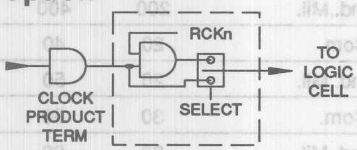


Figure 5

I/O Pin Logic

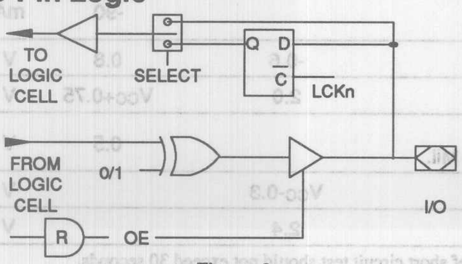


Figure 6

Logic Cell with Buried Sum Term and Register to I/O Cell

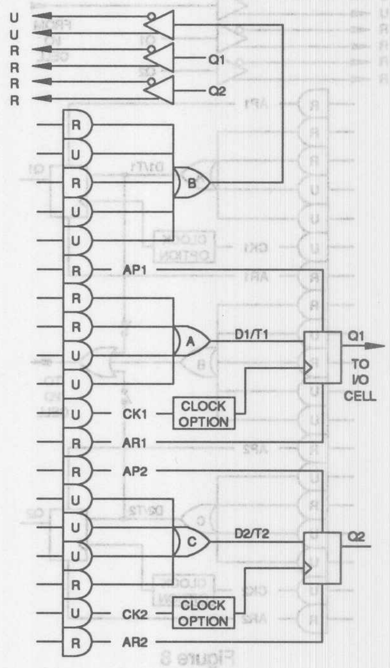


Figure 7

Flip-Flop Clock Options

Each register may be connected to its regional clock to provide fast clock-to-output timing (see Figure 5). In this "synchronous" mode, the clock is one of four input pins, a unique clock pin for each chip quadrant. One product term defines each flip-flop's clock in the "asynchronous" mode.

In the "synchronous" mode, the regional clock is ANDed with the product term. This provides the fast timing of a synchronous clock with the local control of the product term.

I/O Pin Latches

Each I/O pin of the ATV5100 has an input latch which can be individually enabled or disabled (see Figure 6). Each chip quadrant has a unique latch clock. When the latch is inactive, pin input flows directly into the array. When activated, the latch is flow-through when the clock signal is low, and data is captured on the clock's rising edge.

Flip-Flop Types

Each flip-flop in the ATV5100 may be configured as either a T- or D-type flip-flop. A T-type flip-flop can also easily be configured into a JK or SR flip-flop.

Logic Cell, Two Buried Registers, Combinatorial to I/O Cell

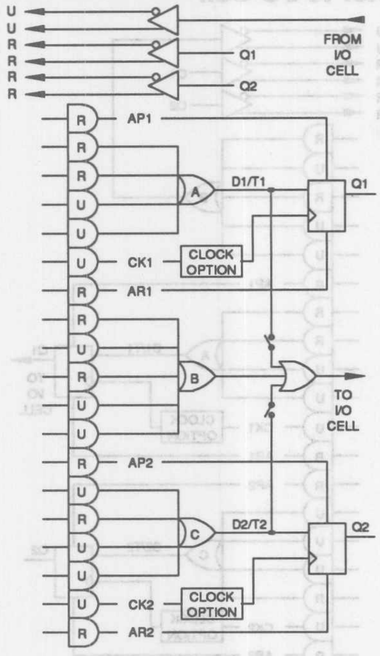


Figure 8

Logic Cell with Combinable Sum Terms, Register to I/O Cell

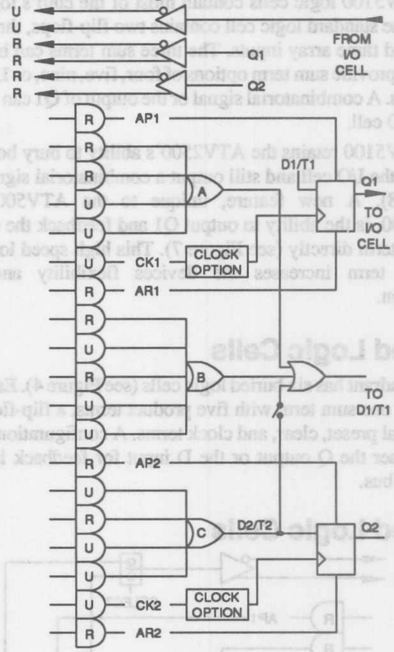


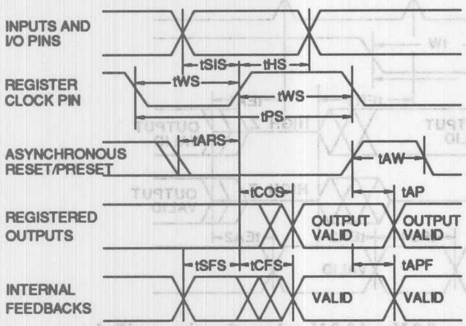
Figure 9

D.C. Characteristics

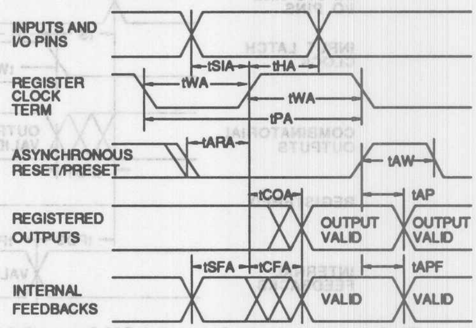
Symbol	Parameter	Condition		Min	Typ	Max	Units
I_{LI}	Input Load Current	V_{IN} = -0.1 V to V_{CC}+1 V				10	µA
I_{LO}	Output Leakage Current	V_{OUT} = -0.1 V to V_{CC}+0.1 V				10	µA
I_{CC}	Power Supply Current ATV5100	V_{CC} = MAX, V_{IN} = GND or V_{CC} Outputs Open	Com.		200	350	mA
			Ind.,Mil.		200	400	mA
I_{CC}	Power Supply Current ATV5100L	V_{CC} = MAX, V_{IN} = GND or V_{CC} Outputs Open	Com.		20	40	mA
			Ind.,Mil.		20	50	mA
I_{CC2}	Clocked Power Supply Current, ATV5100L Only	f = 1 MHz, V_{CC} = MAX Outputs Open	Com.		30	50	mA
			Ind.,Mil.		30	60	mA
I_{OS} (1)	Output Short Circuit Current	V_{OUT} = 0.5 V				-90	mA
V_{IL}	Input Low Voltage			-0.6		0.8	V
V_{IH}	Input High Voltage			2.0		V_{CC}+0.75	V
V_{OL}	Output Low Voltage	V_{IN} = V_{IH} or V_{IL}, I_{OL} = 8 mA Com,Ind; 6 mA Mil.				0.5	V
V_{OH}	Output High Voltage	I_{OH} = -100 µA		V_{CC}-0.3			V
		I_{OH} = -4.0 mA		2.4			V

Note: 1. Not more than one output at a time should be shorted. Duration of short circuit test should not exceed 30 seconds.

A.C. Waveforms (1) Input Pin Clock



A.C. Waveforms (1) Product Term Clock



Notes: 1. Timing measurement reference is 1.5 V. Input AC driving levels are 0.0 V and 3.0 V, unless otherwise specified.

Register A.C. Characteristics, Input Pin Clock

Symbol	Parameter	ATV5100-25		ATV5100/L-30		ATV5100/L-35		Units
		Min	Max	Min	Max	Min	Max	
t_{COS}	Clock to Output		15		20		25	ns
t_{CFS}	Clock to Feedback	0	9	0	12	0	15	ns
t_{SIS}	Input Setup Time(1)	16		17		20		ns
t_{SFS}	Feedback Setup Time(1)	11		13		15		ns
t_{HS}	Hold Time	0		0		0		ns
t_{WS}	Clock Width	10		12		15		ns
t_{PS}	Clock Period	20		25		30		ns
F_{MAXS}	Maximum Frequency (1/t_{PS})		50		40		33	MHz
t_{ARS}	Asynchronous Reset/Preset Recovery Time	20		25		30		ns

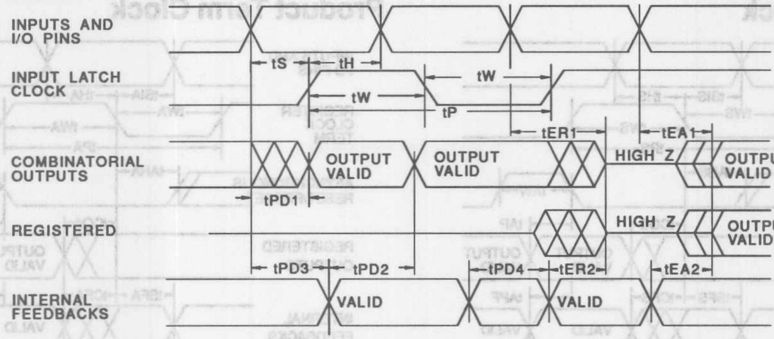
Note: 1. Add 3 ns for Universal Product Terms.

Register A.C. Characteristics, Product Term Clock

Symbol	Parameter	ATV5100-25		ATV5100/L-30		ATV5100/L-35		Units
		Min	Max	Min	Max	Min	Max	
t_{COA}	Clock to Output		25		30		35	ns
t_{CFA}	Clock to Feedback	7	20	10	25	12	27	ns
t_{SIA}	Input Setup Time(1)	10		12		15		ns
t_{SFA}	Feedback Setup Time(1)	5		8		13		ns
t_{HA}	Hold Time	8		10		12		ns
t_{WA}	Clock Width	12		15		15		ns
t_{PA}	Clock Period	25		33		40		ns
F_{MAXA}	Maximum Frequency (1/t_{PA})		40		30		25	MHz
t_{ARA}	Asynchronous Reset/Preset Recovery Time	15		20		25		ns

Note: 1. Add 3 ns for Universal Product Terms.

A.C. Waveforms [1]



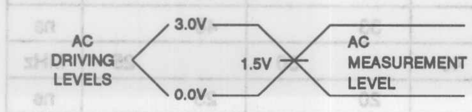
Notes: 1. Timing measurement reference is 1.5 V. Input AC driving levels are 0.0 V and 3.0 V, unless otherwise specified.

A.C. Characteristics

Symbol	Parameter	ATV5100-25 Min	ATV5100-25 Max	ATV5100/L-30 Min	ATV5100/L-30 Max	ATV5100/L-35 Min	ATV5100/L-35 Max	Units
t_{PD1}	Input to Non-Registered Output[1]		25		30		35	ns
t_{PD2}	Feedback to Non-Registered Output[1]		20		25		30	ns
t_{PD3}	Input to Non-Registered Feedback[1]		20		25		30	ns
t_{PD4}	Feedback to Non-Registered Feedback[1]		15		18		22	ns
t_{EA1}	Input to Output Enable		30		35		40	ns
t_{ER1}	Input to Output Disable		30		35		40	ns
t_{EA2}	Feedback to Output Enable		25		30		35	ns
t_{ER2}	Feedback to Output Disable		25		30		35	ns
t_S	Input Latch Setup Time	5		6		7		ns
t_H	Input Latch Hold Time	5		5		5		ns
t_W	Clock Width	10		12		12		ns
t_P	Clock Period	20		25		30		ns
F_{MAX}	Maximum Frequency ($1/t_P$)		50		40		33	MHz
t_{AW}	Asynchronous Reset/Preset Width	15		20		20		ns
t_{AP}	Asynchronous Reset/ Preset to Registered Output		30		35		40	ns
t_{APF}	Asynchronous Reset/ Preset to Registered Feedback		25		30		35	ns

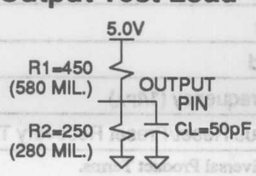
Note: 1. Add 3 ns for Universal Product Terms.

Input Test Waveforms and Measurement Levels



t_R, t_F < 5 ns (10% to 90%)

Output Test Load



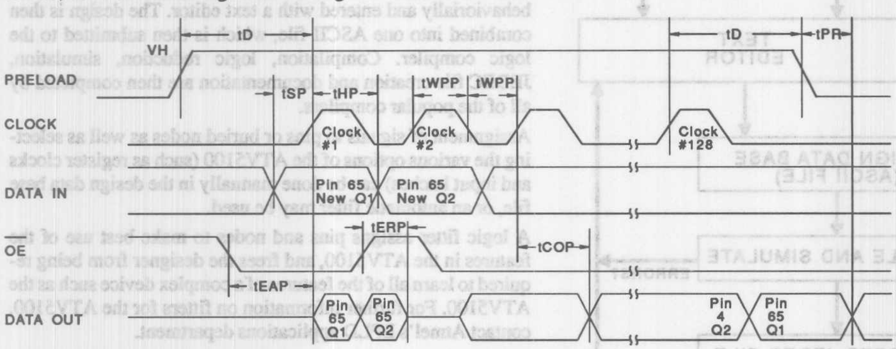
Preload and Observability of Registers

The ATV5100's registers include circuitry to load and unload them serially. This feature simplifies testing. Any state can be forced into the registers to control test sequencing, and all registers may be observed, independent of being buried. A V_{IH} level on the Data In pin will force the appropriate register high; a V_{IL} will force it low, independent of the polarity or other configuration bit settings.

The preload/observe state is entered by placing an 11-V to 14-V signal on pin 68 on the JLCC. When the clock (pin 1) is pulsed high, data (pin 2) is clocked serially through all registers in the device, as in the following table. All register contents are also clocked out of the device on Pin 65 in FIFO fashion. If observability only is required, data out should be connected back to data in. If preload only is required, OE (pin 66) can be held high and data out (pin 65) will remain high impedance.

Any user contemplating the use of register preload/obervability is encouraged to contact Atmel's PLD applications department.

Note: All register clock terms or pins must be low prior to entering the preload/observe state, and low prior to leaving the preload/observe state. Pin 1 must be low prior to entering the preload/observe state.



tD_{MIN} = 100 ns
tSP_{MIN} = 50 ns
tHP_{MIN} = 50 ns
$tWPP_{MIN}$ = 100 ns
tPR_{MIN} = 1000 ns
$tERP_{MAX}$ = 100 ns
$tEAP_{MAX}$ = 100 ns
$tCOP_{MAX}$ = 100 ns

Preload / Observe Register Scan Order

Quadrant	Pin
Quadrant 1	Pin 4 5 6 ••• 15 17 D_{IN} →Q2 →Q1 →B23→Q2 →Q1 →Q2 →Q1 ••• B18 →Q2 →Q1 →Q2 →Q1 →(Quadrant 2)
Quadrant 2 (Quadrant 1)→	Pin 18 19 21 22 ••• 31 Q2 →Q1 →Q2 →Q1 →B17→Q2 →Q1 →Q2 →Q1 ••• B12 →Q2 →Q1 →(Quadrant 3)
Quadrant 3 (Quadrant 2)→	Pin 38 39 40 ••• 49 51 Q2 →Q1 →B11→Q2 →Q1 →Q2 →Q1 ••• B6 →Q2 →Q1 →Q2 →Q1 →(Quadrant 4)
Quadrant 4 (Quadrant 3)→	Pin 52 53 55 56 ••• 65 Q2 →Q1 →Q2 →Q1 →B5 →Q2 →Q1 →Q2 →Q1 ••• B0 →Q2 →Q1 →D_{OUT}

Power Up Reset

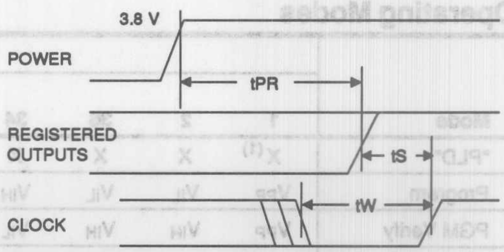
The registers in the ATV5100 are designed to reset during power up. At a point delayed slightly from V_{CC} crossing 3.8 V, all registers will be reset to the low state. The output state will depend on the polarity of the output buffer.

This feature is critical for state machine initialization. However, due to the asynchronous nature of reset and the uncertainty of how V_{CC} actually rises in the system, the following conditions are required:

1) The V_{CC} rise must be monotonic,

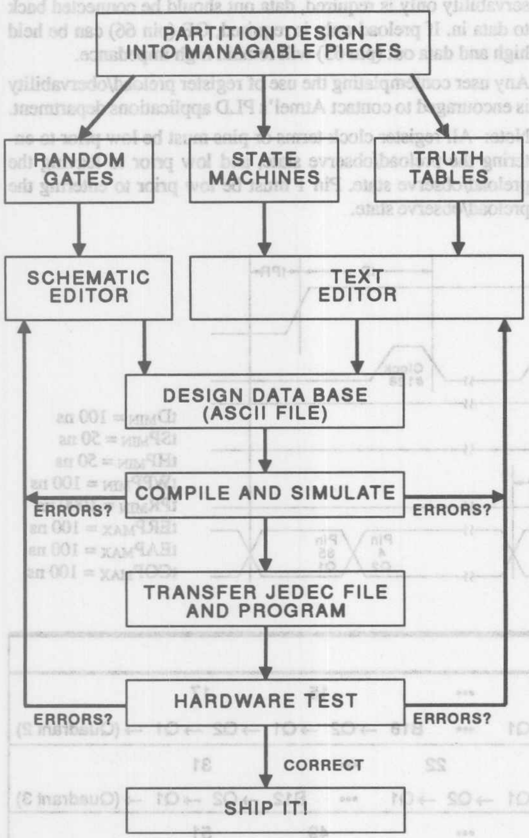
2) After reset occurs, all input and feedback setup times must be met before driving the clock term high, and

3) The signals from which the clock is derived must remain stable during t_{PR}.



Parameter	Description	Min	Typ	Max	Units
t_{PR}	Power-Up Reset Time		600	1000	ns

Design Flow Diagram



Using The ATV5100

The ATV5100's simple, regular architecture means that only simple logic compilers are required to configure the device. No layout or route and place are required. These software tools are readily available from companies such as Data I/O Corporation (Abel™), Logical Devices (Cupl™), and ISDATA (LOGiC™).

The first step in designing a device as complex as the ATV5100 is to partition your design into manageable blocks. These blocks are then allocated proportionally to each of the four quadrants of the ATV5100. Random gates can be described either with boolean equations (a behavioral description) or with a schematic editor. Truth table logic and state machines are best described behaviorially and entered with a text editor. The design is then combined into one ASCII file, which is then submitted to the logic compiler. Compilation, logic reduction, simulation, JEDEC file creation and documentation are then completed by all of the popular compilers.

Assignment of signals to pins or buried nodes as well as selecting the various options of the ATV5100 (such as register clocks and input latches) can be done manually in the design data base file, or an automatic fitter may be used.

A logic fitter assigns pins and nodes to make best use of the features in the ATV5100, and frees the designer from being required to learn all of the features of a complex device such as the ATV5100. For further information on fitters for the ATV5100, contact Atmel's EPLD applications department.

After correcting any syntax and logic errors discovered by the compiler, the JEDEC file is ready to download to an EPLD programmer. These are available from a number of manufacturers. Programmed devices are usually first tested in the programmer with your supplied test vectors. The next step is check out your "custom chip" in the target system. When this hardware debug step is complete, your system is ready to go— all in a matter of hours.

Abel™, Cupl™ and LOGiC™ may be trademarks of others.

Operating Modes

	68-Lead LCC								
Mode	1	2	36	34	68	66	V_{CC} (3,20,37,54)	I/O's	I/O Pin 65
"PLD"	X (1)	X	X	X	X	X	5V	I/O	I/O
Program	V_{PP}	V_{IL}	V_{IL}	V_{IH}	V_{IH}	X	6V	ADD/D_{IN}	ADD
PGM Verify	V_{PP}	V_{IH}	V_{IH}	V_{IL}	V_{IL}	X	6V	ADD/D_{OUT}	ADD
PGM Inhibit	V_{PP}	V_{IH}	V_{IH}	V_{IH}	V_{IH}	X	6V	High Z	High Z
Preload/Observe	⎍	D_{IN}	X	X	V_H (2)	$\overline{OE}$	5V	High Z	D_{OUT}

Notes: 1. X can be V_{IL} or V_{IH}.
2. V_H = 11.0 V to 14.0 V

ATV5100 PLCC/PGA Pin Assignments

PLCC Pin	PGA Pin	Name	PLCC Pin	PGA Pin	Name	PLCC Pin	PGA Pin	Name	PLCC Pin	PGA Pin	Name
1	B6	IN	18	F2	I/O	35	K6	IN	52	F10	I/O
2	A6	IN	19	F1	I/O	36	L6	IN	53	F11	I/O
3	B5	VCC	20	G2	VCC	37	K7	VCC	54	E10	VCC
4	A5	I/O	21	G1	I/O	38	L7	I/O	55	E11	I/O
5	B4	I/O	22	H2	I/O	39	K8	I/O	56	D10	I/O
6	A4	I/O	23	H1	I/O	40	L8	I/O	57	D11	I/O
7	B3	I/O	24	J2	I/O	41	K9	I/O	58	C10	I/O
8	A3	I/O	25	J1	I/O	42	L9	I/O	59	C11	I/O
9	A2	I/O	26	K1	I/O	43	L10	I/O	60	B11	I/O
10	B2	I/O	27	K2	I/O	44	K10	I/O	61	B10	I/O
11	B1	I/O	28	L2	I/O	45	K11	I/O	62	A10	I/O
12	C2	I/O	29	K3	I/O	46	J10	I/O	63	B9	I/O
13	C1	I/O	30	L3	I/O	47	J11	I/O	64	A9	I/O
14	D2	I/O	31	K4	I/O	48	H10	I/O	65	B8	I/O
15	D1	I/O	32	L4	IN	49	H11	I/O	66	A8	IN
16	E2	GND	33	K5	GND	50	G10	GND	67	B7	GND
17	E1	I/O	34	L5	IN	51	G11	I/O	68	A7	IN

Pin Capacitance (f = 1 MHz, T = 25°C) (1)

	Typ	Max	Units	Conditions
C_{IN}	6	8	pF	$V_{IN} = 0$ V
C_{OUT}	8	12	pF	$V_{OUT} = 0$ V

Note: 1. Typical values for nominal supply voltage. This parameter is only sampled and is not 100% tested.

Security Fuse Usage

A single fuse is provided to prevent unauthorized copying of the ATV5100 fuse patterns. Once programmed, all outputs appear programmed during verify. The security fuse should be programmed last (after verifying all other programmed bits), as its effect is immediate.

The security fuse also inhibits preload and observability.

Erasure Characteristics

The entire memory array of an ATV5100 is erased after exposure to ultraviolet light at a wavelength of 2537 Å. Complete erasure is assured after a minimum of 20 minutes exposure using 12,000 μW/cm^2 intensity lamps spaced one inch away from the chip. Minimum erase time for lamps at other intensity ratings can be calculated from the minimum integrated erasure dose of 15 W·sec/cm^2. To prevent unintentional erasure, an opaque label is recommended to cover the clear window on any UV erasable PLD which will be subjected to continuous fluorescent indoor lighting or sunlight.

Ordering Information

t_{PD} (ns)	t_{COS} (ns)	f_{MAX} (MHz)	Ordering Code	Package	Operation Range
25	15	50	ATV5100-25JC ATV5100-25KC ATV5100-25UC	68J 68KW 68UW	Commercial (0°C to 70°C)
			ATV5100-25KI ATV5100-25UI	68KW 68UW	Industrial (-40°C to 85°C)
			ATV5100-25KM ATV5100-25UM	68KW 68UW	Military (-55°C to 125°C)
30	20	40	ATV5100-30JC ATV5100-30KC ATV5100-30UC	68J 68KW 68UW	Commercial (0°C to 70°C)
			ATV5100-30KI ATV5100-30UI	68KW 68UW	Industrial (-40°C to 85°C)
			ATV5100-30KM ATV5100-30UM	68KW 68UW	Military (-55°C to 125°C)
			ATV5100-30KM/883 ATV5100-30UM/883	68KW 68UW	Military/883C Class B, Fully Compliant (-55°C to 125°C)
35	25	33	ATV5100-35JC ATV5100-35KC ATV5100-35UC	68J 68KW 68UW	Commercial (0°C to 70°C)
			ATV5100-35KI ATV5100-35UI	68KW 68UW	Industrial (-40°C to 85°C)
			ATV5100-35KM ATV5100-35UM	68KW 68UW	Military (-55°C to 125°C)
			ATV5100-35KM/883 ATV5100-35UM/883	68KW 68UW	Military/883C Class B, Fully Compliant (-55°C to 125°C)

Ordering Information

t_{PD} (ns)	t_{COS} (ns)	f_{MAX} (MHz)	Ordering Code	Package	Operation Range
30	20	40	ATV5100L-30JC ATV5100L-30KC ATV5100L-30UC	68J 68KW 68UW	Commercial (0°C to 70°C)
35	25	33	ATV5100L-35JC ATV5100L-35KC ATV5100L-35UC	68J 68KW 68UW	Commercial (0°C to 70°C)
			ATV5100L-35KI ATV5100L-35UI	68KW 68UW	Industrial (-40°C to 85°C)
			ATV5100L-35KM ATV5100L-35UM	68KW 68UW	Military (-55°C to 125°C)
			ATV5100L-35KM/883 ATV5100L-35UM/883	68KW 68UW	Military/883C Class B, Fully Compliant (-55°C to 125°C)

Package Type	
68J	68 Lead, Plastic J-Leaded Chip Carrier OTP (PLCC)
68KW	68 Lead, Windowed, Ceramic J-Leaded Chip Carrier (JLCC)
68UW	68 Pin, Windowed, Ceramic Pin Grid Array (PGA)

Contents

Section 2 Field Programmable Gate Arrays (FPGAs)

Field Programmable Gate Arrays

Features

- **High Performance**
  - **System Speeds to 70 MHz**
  - **Flip-Flop Toggle Rates to 250 MHz**
- **Symmetrical Architecture**
  - **Thousands of Registers**
  - **Flexible Busing Network**
  - **Predictable Timing Delays**
- **100% Factory-Tested**
- **Cache Logic™ Design**
  - **Complete/Partial In-System Reconfiguration**
  - **No Loss of Data or Machine State**
  - **Adaptive Hardware**
- **Very Low Power Consumption**
  - **Standby Current of 500 µA**
  - **Typical Operating Current of 50 to 70 mA**
- **Programmable Clock Options**
  - **Independently Controlled Column Clocks**
  - **Independently Controlled Column Resets**
  - **Clock Skew Less Than 1 ns Across Chip**
- **Independently Configurable I/O**
  - **High-I/O Versions for Low-Density Devices**
  - **TTL/CMOS Input Thresholds**
  - **Open Collector/Tri-state Outputs**
  - **Programmable Slew-Rate Control**
  - **I/O Drive to 12 mA (Combinable to 48 mA)**
  - **Registered/Direct Inputs and Outputs**

Description

AT6000 Series Field Programmable Gate Arrays (FPGAs) provide the density and performance of custom gate arrays without the prototyping and debugging delays associated with mask-programmed devices.

Supporting system speeds of up to 70 MHz and using a typical operating current of 50 to 70 mA, AT6000 Series devices are ideal for high-speed, data-path designs. These FPGAs are designed to be reprogrammable, so they reduce design risk, shorten design cycles and speed time to market.

The patented AT6000 Series architecture employs a symmetrical grid of small yet powerful cells connected to a flexible busing network. Independently controlled clocks and resets govern every column of cells. The array is surrounded by programmable I/O.

Devices range in size from 2,000 to 10,000 usable gates. Pin locations are consistent throughout the Series for easy design migration. High-I/O versions are available for the lower gate count devices.

AT6000 Series FPGAs utilize a reliable 0.8-µm single-poly, double-metal CMOS process and are 100% factory-tested.

continued on next page

AT6000 Series Field Programmable Gate Arrays

Device	AT6002	AT6003	AT6005	AT6010
Capacity, Gates	2,000	3,000	5,000	10,000
Cells	1,024	1,600	3,136	6,400
Registers (maximum)	1,024	1,600	3,136	6,400
I/O (maximum)	96	120	108	173
Typ. Operating Current (mA)	30	45	80	170
Cell Rows x Columns	32 x 32	40 x 40	56 x 56	80 x 80

Description (Continued)

Atmel's PC-based Integrated Development System is used to create AT6000 Series designs. Other platforms are in development.

The Atmel architecture was developed to provide the highest levels of performance, functional density and design flexibility in an FPGA. The cells in the Atmel array are small, very efficient and contain the most important and most commonly used logic and wiring functions. The cell's small size leads to arrays with large numbers of cells, greatly multiplying the functionality in each cell. A simple, high-speed busing network provides fast, efficient communication over medium and long distances.

The Symmetrical Array

At the heart of the Atmel architecture is a symmetrical array of identical cells (Figure 1). The array is continuous and completely uninterrupted from one edge to the other, except for bus *repeaters* spaced every eight cells (Figure 2).

In addition to logic and storage, cells can also be used as wires to connect functions together over short distances and are useful for routing in tight spaces. Buses support fast, efficient communication over medium and long distances.

The Busing Network

There are two kinds of buses: local and express (see Figures 2 and 3).

Local buses are the link between the array of cells and the busing network. There are two local buses— North-South 1 and 2 (NS1 and NS2)— for every column of cells, and two local buses— East-West 1 and 2 (EW1 and EW2)— for every row of cells. Each local bus is connected to every cell in its column or row, thus providing every cell in the array with read/write access to two North-South and two East-West buses.

Each cell, in addition, provides the ability to route a signal on a 90° turn between the NS1 bus and EW1 bus and between the NS2 bus and EW2 bus.

Express buses are not connected directly to cells, and thus provide higher speeds. They are the fastest way to cover long, straight-line distances within the array.

Each express bus is paired with a local bus, so there are two express buses for every column and two express buses for every row of cells.

Connective units, called repeaters, spaced every eight cells, divide each bus, both local and express, into segments spanning eight cells. Repeaters are aligned in rows and columns thereby partitioning the array into 8 x 8 sectors of cells. Each repeater is associated with a local/express pair, and on each side of the repeater are connections to a local-bus segment and an express-bus segment. The repeater can be programmed to provide any one of twenty-one connecting functions. These functions are

Figure 1. Symmetrical Array Surrounded by I/O

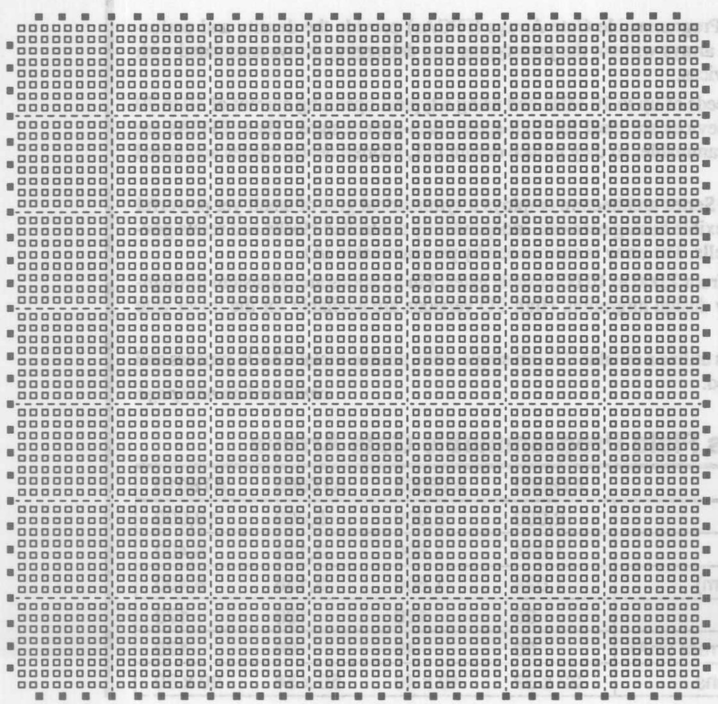


Figure 2. Busing Network

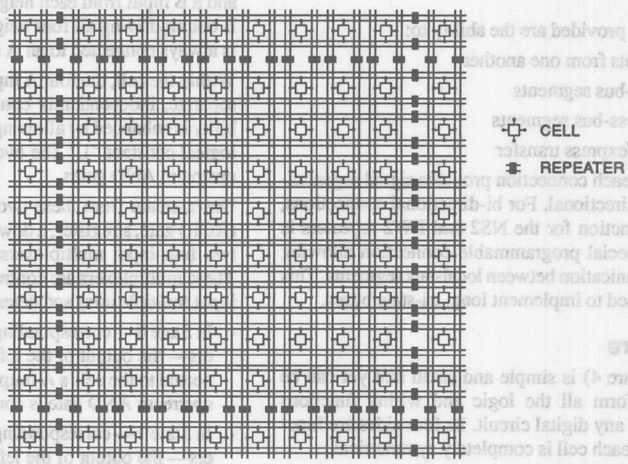
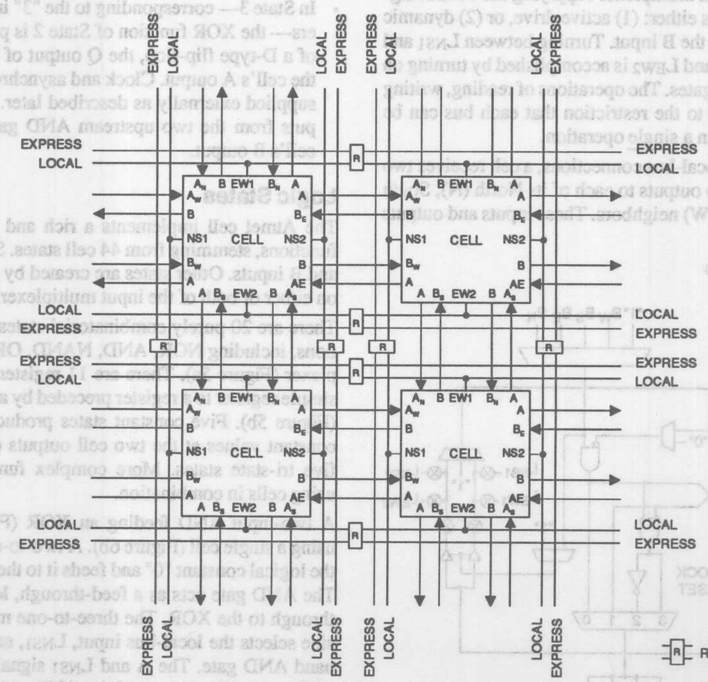


Figure 3. Cell-to-Cell and Bus-to-Bus Connections



symmetric with respect to both the two repeater sides and the two types of buses.

Among the functions provided are the ability to:

- Isolate bus segments from one another
- Connect two local-bus segments
- Connect two express-bus segments
- Implement a local/express transfer

In all of these cases, each connection provides signal regeneration and is thus uni-directional. For bi-directional connections, the basic repeater function for the NS2 and EW2 repeaters is augmented with a special programmable connection allowing bi-directional communication between local-bus segments. This option is primarily used to implement long, tri-state buses.

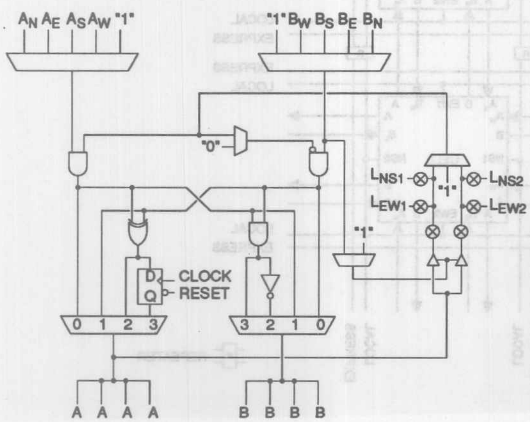
The Cell Structure

The Atmel cell (Figure 4) is simple and small and yet can be programmed to perform all the logic and wiring functions needed to implement any digital circuit. Its four sides are functionally identical, so each cell is completely symmetrical.

Read/write access to the four local buses— NS1, EW1, NS2 and EW2— is controlled, in part, by four bi-directional pass gates connected directly to the buses. To read a local bus, the pass gate for that bus is turned on and the three-input multiplexer is set accordingly. To write to a local bus, the pass gate for that bus and the pass gate for the associated tri-state driver are both turned on. The two-input multiplexer supplying the control signal to the drivers permits either: (1) active drive, or (2) dynamic tri-stating controlled by the B input. Turning between L_{NS1} and L_{EW1} or between L_{NS2} and L_{EW2} is accomplished by turning on the two associated pass gates. The operations of reading, writing and turning are subject to the restriction that each bus can be involved in no more than a single operation.

In addition to the four local-bus connections, a cell receives two inputs and provides two outputs to each of its North (N), South (S), East (E) and West (W) neighbors. These inputs and outputs are divided into two classes: "A" and "B." There is an A input and a B input from each neighboring cell and an A output and a B output driving all four neighbors. Between cells, an A output is always connected to an A input and a B output to a B input.

Figure 4. Cell Structure



Within the cell, the four A inputs and the four B inputs enter two separate, independently configurable multiplexers. Cell flexibility is enhanced by allowing each multiplexer to select also the logical constant "1." The two multiplexer outputs enter the two upstream AND gates.

Downstream from these two AND gates are an Exclusive-OR (XOR) gate, a register, a downstream AND gate, an inverter and two four-input multiplexers producing the A and B outputs. These multiplexers are controlled in tandem (unlike the A and B input multiplexers) and determine the function of the cell.

- In State 0— corresponding to the "0" inputs of the multiplexers— the output of the left-hand upstream AND gate is connected to the cell's A output, and the output of the right-hand upstream AND gate is connected to the cell's B output.
- In State 1— corresponding to the "1" inputs of the multiplexers— the output of the left-hand upstream AND gate is connected to the cell's B output, the output of the right-hand upstream AND gate is connected to the cell's A output.
- In State 2— corresponding to the "2" inputs of the multiplexers— the XOR of the outputs from the two upstream AND gates is provided to the cell's A output, while the NAND of these two outputs is provided to the cell's B output.
- In State 3— corresponding to the "3" inputs of the multiplexers— the XOR function of State 2 is provided to the D input of a D-type flip-flop, the Q output of which is connected to the cell's A output. Clock and asynchronous reset signals are supplied externally as described later. The AND of the outputs from the two upstream AND gates is provided to the cell's B output.

Logic States

The Atmel cell implements a rich and powerful set of logic functions, stemming from 44 cell states. Some states use both A and B inputs. Other states are created by selecting the "1" input on either or both of the input multiplexers.

There are 20 purely combinatorial states with a range of functions, including NOR, AND, NAND, OR and two-input multiplexer (Figure 5a). There are 11 register states ranging from a simple register to a register preceded by a two-input multiplexer (Figure 5b). Five constant states produce all combinations of constant values at the two cell outputs (Figure 5c). There are five tri-state states. More complex functions are created by using cells in combination.

A two-input AND feeding an XOR (Figure 6a) is produced using a single cell (Figure 6b). A two-to-one multiplexer selects the logical constant "0" and feeds it to the right-hand AND gate. The AND gate acts as a feed-through, letting the B input pass through to the XOR. The three-to-one multiplexer on the right side selects the local-bus input, L_{NS1}, and passes it to the left-hand AND gate. The A and L_{NS1} signals are the inputs to the AND gate. The output of the AND gate feeds into the XOR, producing the logic state (A•L) XOR B.

Figure 5a. Combinatorial States

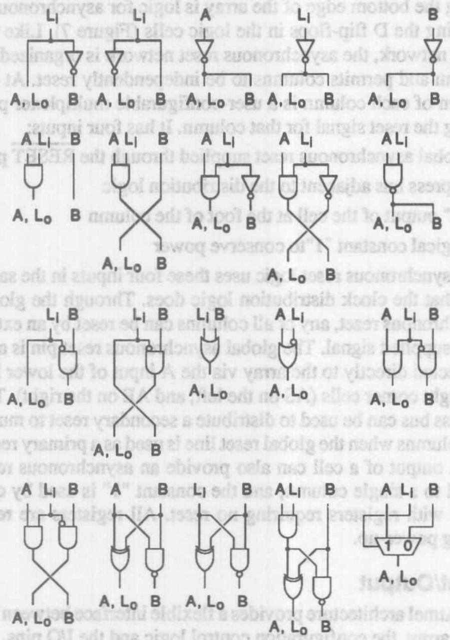


Figure 5b. Register States

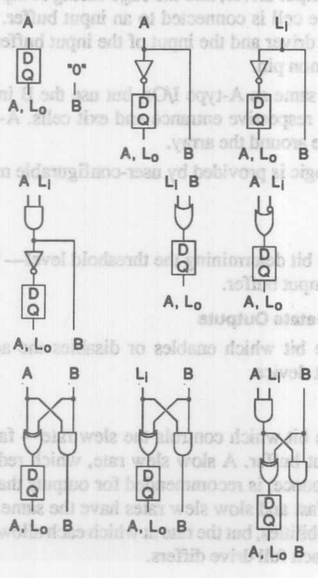
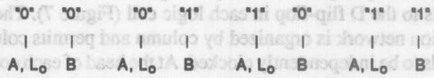


Figure 5c. Constant States



2

Figure 6a. Two-Input AND Feeding XOR

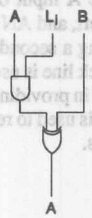
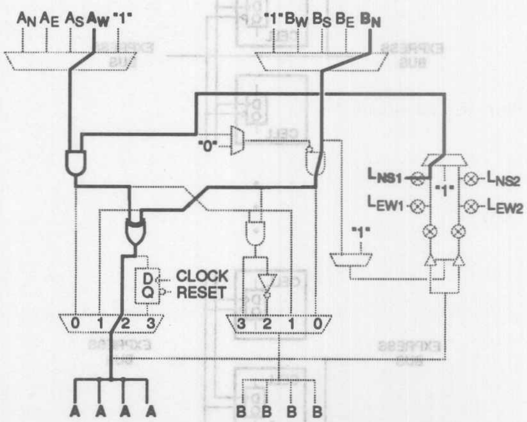


Figure 6b. Cell Configuration: (A • L) XOR B



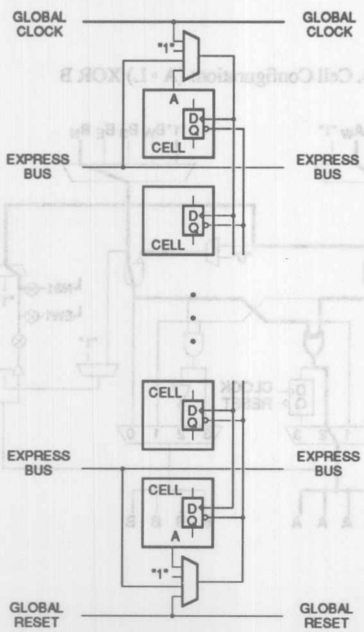
Clock Distribution

Along the top edge of the array is logic for distributing clock signals to the D flip-flop in each logic cell (Figure 7). The distribution network is organized by column and permits columns of cells to be independently clocked. At the head of each column is a user-configurable multiplexer providing the clock signal for that column. It has four inputs:

- Global clock supplied through the CLOCK pin
- Express bus adjacent to the distribution logic
- "A" output of the cell at the head of the column
- Logical constant "1" to conserve power (no clock)

Through the global clock, the network provides low-skew distribution of an externally supplied clock to any or all the columns of the array. The global clock pin is also connected directly to the array via the A input of the upper left and right corner cells (AW on the left, and AN on the right). The express bus is useful in distributing a secondary clock to multiple columns when the global clock line is used as a primary clock. The A output of a cell is useful in providing a clock signal to a single column. The constant "1" is used to reduce power dissipation in columns using no registers.

Figure 7. Column Clock and Column Reset



Asynchronous Reset

Along the bottom edge of the array is logic for asynchronously resetting the D flip-flops in the logic cells (Figure 7). Like the clock network, the asynchronous reset network is organized by column and permits columns to be independently reset. At the bottom of each column is a user-configurable multiplexer providing the reset signal for that column. It has four inputs:

- Global asynchronous reset supplied through the $\overline{\text{RESET}}$ pin
- Express bus adjacent to the distribution logic
- "A" output of the cell at the foot of the column
- Logical constant "1"to conserve power

The asynchronous reset logic uses these four inputs in the same way that the clock distribution logic does. Through the global asynchronous reset, any or all columns can be reset by an externally supplied signal. The global asynchronous reset pin is also connected directly to the array via the A input of the lower left and right corner cells (AS on the left, and AE on the right). The express bus can be used to distribute a secondary reset to multiple columns when the global reset line is used as a primary reset, the A output of a cell can also provide an asynchronous reset signal to a single column, and the constant "1" is used by columns with registers requiring no reset. All registers are reset during power-up.

Input/Output

The Atmel architecture provides a flexible interface between the logic array, the configuration control logic and the I/O pins.

Two adjacent cells— an "exit" and an "entrance" cell— on the perimeter of the logic array are associated with each I/O pin.

There are two types of I/Os: A-type (Figure 8a) and B-type (Figure 8b). For A-type I/Os, the edge-facing A output of an exit cell is connected to an output driver, and the edge-facing A input of the adjacent entrance cell is connected to an input buffer. The output of the output driver and the input of the input buffer are connected to a common pin.

B-type I/Os are the same as A-type I/Os, but use the B inputs and outputs of their respective entrance and exit cells. A- and B-type I/Os alternate around the array.

Control of the I/O logic is provided by user-configurable memory bits.

TTL/CMOS Inputs

A user-configurable bit determining the threshold level— TTL or CMOS— of the input buffer.

Open Collector/Tri-state Outputs

A user-configurable bit which enables or disables the active pull-up of the output device.

Slew Rate Control

A user-configurable bit which controls the slew rate— fast or slow— of the output buffer. A slow slew rate, which reduces noise and ground bounce, is recommended for outputs that are not speed-critical. Fast and slow slew rates have the same DC-current sinking capabilities, but the rate at which each allows the output devices to reach full drive differs.

Pull-up

A user-configurable bit controlling the pull-up transistor in the I/O pin. It's primary function is to provide a logical "1" to unused input pins. When on, it is roughly equivalent to a 25K resistor to V_{CC}.

Enable Select

User-configurable bits determining the output-enable for the output driver. The output driver can be static, always on, always off, or dynamically controlled by a signal generated in the array. Four options are available from the array: (1) the control is low and always driving; (2) the control is high and never driving; (3) the control is connected to a vertical local bus associated with the output cell; and (4) the control is connected to a horizontal local bus associated with the output cell. The power-up default is never driving.

In addition to the functionality provided by the I/O logic, the entrance and exit cells provide the ability to register both inputs and outputs. Also, these perimeter cells (unlike interior cells) are connected directly to express buses: the edge-facing A and B outputs of the entrance cell are connected to express buses, as are the edge-facing A and B inputs of the exit cell. These buses are perpendicular to the edge, and provide a rapid means of bringing I/O signals to and from the array interior and the opposite edge of the chip.

Chip Configuration

The Integrated Development System generates the SRAM bit pattern required to configure a AT6000 Series device. A PC parallel port, microprocessor or EPROM can be used to download configuration patterns.

Users select from several configuration modes. Many factors, including board area, configuration speed and the number of designs implemented in parallel can influence the user's final choice.

Configuration is controlled by dedicated configuration pins and dual-function pins that double as I/O pins when the device is in operation. The number of dual-function pins required for each mode varies.

The devices can be partially reconfigured while in operation. Portions of the device not being modified remain operational during configuration. Simultaneous configuration of more than one device is also possible. Full configuration takes as little as a millisecond, partial configuration is even faster.

Refer to the Pin Function Description section following for a brief summary of the pins used in configuration. For more information about configuration, refer to the AT6000 Series Configuration data sheet.

2

Figure 8a. A-Type I/O Logic

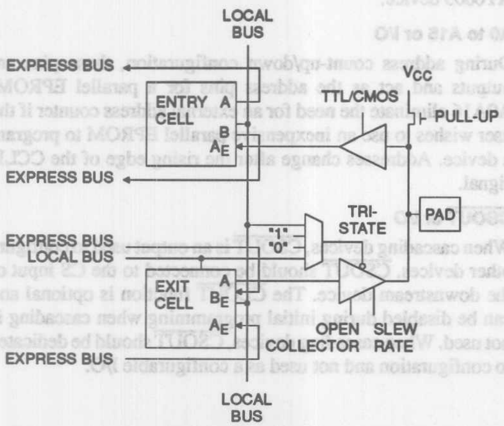
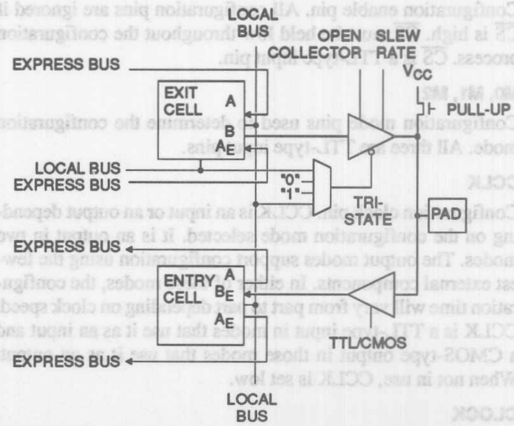


Figure 8b. B-Type I/O Logic



Pin Function Description

This section provides abbreviated descriptions of the various AT6000 Series pins. For more complete descriptions, refer to the AT6000 Series Configuration data sheet.

Pinout tables for the AT6002 and AT6005 devices follow.

Power Pins

V_{CC}, V_{DD}, GND, V_{SS}

V_{CC} and GND are the I/O supply pins, V_{DD} and V_{SS} are the internal logic supply pins. V_{CC} and V_{DD} should be tied to the same trace on the printed circuit board. GND and V_{SS} should be tied to the same trace on the printed circuit board.

Input/Output Pins

All I/O pins can be used in the same way (refer to the I/O section of the architecture description). Some I/O pins are dual-function pins used during configuration of the array. When not being used for configuration, dual-function I/Os are fully functional as normal I/O pins. On initial power-up, all I/Os are configured as TTL inputs with a pull-up.

Dedicated Timing and Control Pins

$\overline{\text{CON}}$

Configuration in process pin. After power-up, $\overline{\text{CON}}$ remains low until power-up initialization is complete. $\overline{\text{CON}}$ is an open collector signal. After power-up initialization, forcing $\overline{\text{CON}}$ low begins the configuration process.

$\overline{\text{CS}}$

Configuration enable pin. All configuration pins are ignored if $\overline{\text{CS}}$ is high. $\overline{\text{CS}}$ must be held low throughout the configuration process. $\overline{\text{CS}}$ is a TTL-type input pin.

M0, M1, M2

Configuration mode pins used to determine the configuration mode. All three are TTL-type input pins.

CCLK

Configuration clock pin. CCLK is an input or an output depending on the configuration mode selected. It is an output in two modes. The output modes support configuration using the fewest external components. In either of these modes, the configuration time will vary from part to part depending on clock speed. CCLK is a TTL-type input in modes that use it as an input and a CMOS-type output in those modes that use it as an output. When not in use, CCLK is set low.

CLOCK

External logic source used to drive the internal global clock line. Registers toggle on the rising edge of CLOCK. The CLOCK signal is neither used nor affected by the configuration modes. It is always a TTL input.

$\overline{\text{RESET}}$

Array register asynchronous reset. $\overline{\text{RESET}}$ drives the internal global reset. The $\overline{\text{RESET}}$ signal is neither used nor affected by the configuration modes. It is always a TTL input.

Dual-Function Pins

When $\overline{\text{CON}}$ is high, dual-function I/O pins act as device I/Os; when $\overline{\text{CON}}$ is low, dual-function pins are used as configuration control or data signals as determined by the configuration modes. Care must be taken when using these pins to ensure that configuration activity does not interfere with other circuitry associated with the pin's net.

D0 or I/O

Serial configuration modes use D0 as the serial data input pin. Parallel configuration modes use D0 as the least-significant bit. Input data must meet setup and hold requirements with respect to the rising edge of CCLK.

D1 to D7 or I/O

Parallel configuration modes use these pins as inputs. Serial configuration modes do not use them. Data must meet setup and hold requirements with respect to the rising edge of CCLK.

$\overline{\text{CEN}}$ or I/O

During address count-up/down configuration, $\overline{\text{CEN}}$ is an output. $\overline{\text{CEN}}$ can be used as the output enable of a parallel EPROM. In this case, it should be configured as a constant high, and not used as a configurable I/O pin. $\overline{\text{CEN}}$ is only available on the AT6005 device.

A0 to A16 or I/O

During address count-up/down configuration, these pins are outputs and act as the address pins for a parallel EPROM. A0A16 eliminate the need for an external address counter if the user wishes to use an inexpensive parallel EPROM to program a device. Addresses change after the rising edge of the CCLK signal.

$\overline{\text{CSOUT}}$ or I/O

When cascading devices, $\overline{\text{CSOUT}}$ is an output used to configure other devices. $\overline{\text{CSOUT}}$ should be connected to the CS input of the downstream device. The $\overline{\text{CLOUT}}$ function is optional and can be disabled during initial programming when cascading is not used. When cascading devices, $\overline{\text{CSOUT}}$ should be dedicated to configuration and not used as a configurable I/O.

$\overline{\text{CHECK}}$ or I/O

During configuration, $\overline{\text{CHECK}}$ is an input that can be used to enable the data check function at the beginning of a configuration cycle. No data is written to the device while $\overline{\text{CHECK}}$ is low. Instead, the configuration file being applied to D0-D7 is compared with the current contents of the internal configuration RAM. If a mismatch is detected between the data being loaded and the data already in the RAM, the $\overline{\text{ERR}}$ pin goes low. The $\overline{\text{CHECK}}$ function is optional and can be disabled during initial programming.

$\overline{\text{ERR}}$ or I/O

During configuration, $\overline{\text{ERR}}$ is an output. When the $\overline{\text{CHECK}}$ function is activated and a mismatch is detected between the current configuration data stream and the data already loaded in the configuration RAM, $\overline{\text{ERR}}$ goes low. The $\overline{\text{ERR}}$ output is a registered signal. Once a mismatch is found, the signal is set and is only reset after the configuration cycle is complete. $\overline{\text{ERR}}$ is also asserted for configuration file errors. The $\overline{\text{ERR}}$ function is optional and can be disabled during initial programming.

AT6002 Pinout Assignment [(1)]

84-Pin PLCC

- 64 I/O
- 8 Fixed-Function
- 28 Dual-Function
- 4 VCC
- 4 GND
- 2 VDD
- 2 VSS

100-Pin TQFP

- 80 I/O
- 8 Fixed-Function
- 28 Dual-Function
- 4 VCC
- 4 GND
- 2 VDD
- 2 VSS

132-Pin PQFP

- 96 I/O
- 8 Fixed-Function
- 28 Dual-Function
- 4 VCC
- 8 GND
- 2 VDD
- 2 VSS

Pin Name	84-Pin PLCC	100-Pin TQFP	132-Pin PQFP
CLOCK	1	88	1
A15 or I/O	2	89	2
A14 or I/O	3	90	3
VCC	4	91	4
A13 or I/O	5	92	5
I/O	*	*	6
A12 or I/O	6	93	7
I/O	*	94	8
A11 or I/O	7	95	9
*	*	*	10
GND	*	*	11
A10 or I/O	8	96	12
I/O	*	97	13
A9 or I/O	9	98	14
I/O	*	*	15
A8 or I/O	10	99	16
M0	11	100	17
A7 or I/O	12	1	18
*	*	*	19
A6 or I/O	13	2	20
I/O	*	*	21
A5 or I/O	14	3	22
I/O	*	4	23
A4 or I/O	15	5	24
I/O	*	*	25
A3 or I/O	16	6	26
I/O	*	7	27
A2 or I/O	17	8	28
*	*	*	29
GND	18	9	30
VSS	19	10	31
A1 or I/O	20	11	32
A0 or I/O	21	12	33
D7 or I/O	22	13	34
D6 or I/O	23	14	35
D5 or I/O	24	15	36
VDD	25	16	37
VCC	26	17	38
I/O	*	*	39
D4 or I/O	27	18	40
I/O	*	19	41
D3 or I/O	28	20	42
*	*	*	43
GND	*	*	44

Pin Name	84-Pin PLCC	100-Pin TQFP	132-Pin PQFP
D2 or I/O	29	21	45
I/O	*	22	46
D1 or I/O	30	23	47
I/O	*	*	48
D0 or I/O	31	24	49
CCLK	32	25	50
$\overline{\text{CON}}$	33	26	51
I/O	34	27	52
*	*	*	53
$\overline{\text{CSOUT}}$ or I/O	35	28	54
I/O	*	*	55
I/O	36	29	56
I/O	*	30	57
$\overline{\text{CHECK}}$ or I/O	37	31	58
I/O	*	*	59
$\overline{\text{ERR}}$ or I/O	38	32	60
I/O	*	33	61
I/O	39	34	62
*	*	*	63
GND	40	35	64
I/O	41	36	65
I/O	42	37	66
$\overline{\text{CS}}$	43	38	67
I/O	44	39	68
I/O	45	40	69
VCC	46	41	70
I/O	47	42	71
I/O	*	*	72
I/O	48	43	73
I/O	*	44	74
I/O	49	45	75
*	*	*	76
GND	*	*	77
I/O	50	46	78
I/O	*	47	79
I/O	51	48	80
I/O	*	*	81
I/O	52	49	82
$\overline{\text{RESET}}$	53	50	83
I/O	54	51	84
*	*	*	85
I/O	55	52	86
I/O	*	*	87
I/O	56	53	88

Pin Name	84-Pin PLCC	100-Pin TQFP	132-Pin PQFP
I/O	*	54	89
I/O	57	55	90
I/O	*	*	91
I/O	58	56	92
I/O	*	57	93
I/O	59	58	94
*	*	*	95
GND	60	59	96
VSS	61	60	97
I/O	62	61	98
I/O	63	62	99
I/O	64	63	100
I/O	65	64	101
I/O	66	65	102
VDD	67	66	103
VCC	68	67	104
I/O	*	*	105
I/O	69	68	106
I/O	*	69	107
I/O	70	70	108
*	*	*	109
GND	*	*	110
I/O	71	71	111
I/O	*	72	112
I/O	72	73	113
I/O	*	*	114
I/O	73	74	115
M2	74	75	116
M1	75	76	117
I/O	76	77	118
*	*	*	119
I/O	77	78	120
I/O	*	*	121
I/O	78	79	122
I/O	*	80	123
I/O	79	81	124
I/O	*	*	125
I/O	80	82	126
I/O	*	83	127
I/O	81	84	128
*	*	*	129
GND	82	85	130
I/O	83	86	131
A16 or I/O	84	87	132

Note: 1. * Indicates unconnected package pin.

AT6005 Pinout Assignment [1]

84-Pin PLCC

- 64 I/O
- 8 Fixed-Function
- 29 Dual-Function
- 4 VCC
- 4 GND
- 2 VDD
- 2 VSS

100-Pin TQFP [2]

- 80 I/O
- 8 Fixed-Function
- 29 Dual-Function
- 4 VCC
- 4 GND
- 2 VDD
- 2 VSS

132-Pin PQFP

- 108 I/O
- 8 Fixed-Function
- 29 Dual-Function
- 4 VCC
- 8 GND
- 2 VDD
- 2 VSS

144-Pin PQFP

- 108 I/O
- 8 Fixed-Function
- 29 Dual-Function
- 4 VCC
- 8 GND
- 2 VDD
- 2 VSS

Pin Name	84-Pin PLCC	100-Pin TQFP	132-Pin PQFP	144-Pin TQFP
CLOCK	1	88	1	127
A15 or I/O	2	89	2	128
A14 or I/O	3	90	3	129
VCC	4	91	4	130
A13 or I/O	5	92	5	131
I/O	*	*	6	132
A12 or I/O	6	93	7	133
I/O	*	94	8	134
A11 or I/O	7	95	9	135
I/O	*	*	10	136
GND	*	*	11	137
A10 or I/O	8	96	12	138
I/O	*	97	13	139
A9 or I/O	9	98	14	140
I/O	*	*	15	141
N/C	*	*	*	142
A8 or I/O	10	99	16	143
M0	11	100	17	144
A7 or I/O	12	1	18	1
N/C	*	*	*	2
I/O	*	*	19	3
A6 or I/O	13	2	20	4
I/O	*	*	21	5
A5 or I/O	14	3	22	6
I/O	*	4	23	7
A4 or I/O	15	5	24	8
I/O	*	*	25	9
A3 or I/O	16	6	26	10
I/O	*	7	27	11
A2 or I/O	17	8	28	12
I/O	*	*	29	13
GND	18	9	30	14
VSS	19	10	31	15
A1 or I/O	20	11	32	16
N/C	*	*	*	17
A0 or I/O	21	12	33	18
D7 or I/O	22	13	34	19
D6 or I/O	23	14	35	20
D5 or I/O	24	15	36	21
VDD	25	16	37	22
VCC	26	17	38	23
I/O	*	*	39	24
D4 or I/O	27	18	40	25
I/O	*	19	41	26
D3 or I/O	28	20	42	27
I/O	*	*	43	28
GND	*	*	44	29
D2 or I/O	29	21	45	30

Pin Name	84-Pin PLCC	100-Pin TQFP	132-Pin PQFP	144-Pin TQFP
I/O	*	22	46	31
D1 or I/O	30	23	47	32
I/O	*	*	48	33
N/C	*	*	*	34
D0 or I/O	31	24	49	35
CCLK	32	25	50	36
$\overline{\text{CON}}$	33	26	51	37
$\overline{\text{CEN}}$ or I/O	34	27	52	38
N/C	*	*	*	39
I/O	*	*	53	40
$\overline{\text{CSOUT}}$ or I/O	35	28	54	41
I/O	*	*	55	42
I/O	36	29	56	43
I/O	*	30	57	44
$\overline{\text{CHECK}}$ or I/O	37	31	58	45
I/O	*	*	59	46
$\overline{\text{ERR}}$ or I/O	38	32	60	47
I/O	*	33	61	48
I/O	39	34	62	49
I/O	*	*	63	50
GND	40	35	64	51
I/O	41	36	65	52
N/C	*	*	*	53
I/O	42	37	66	54
$\overline{\text{CS}}$	43	38	67	55
I/O	44	39	68	56
I/O	45	40	69	57
VCC	46	41	70	58
I/O	47	42	71	59
I/O	*	*	72	60
I/O	48	43	73	61
I/O	*	44	74	62
I/O	49	45	75	63
I/O	*	*	76	64
GND	*	*	77	65
I/O	50	46	78	66
I/O	*	47	79	67
I/O	51	48	80	68
I/O	*	*	81	69
N/C	*	*	*	70
I/O	52	49	82	71
$\overline{\text{RESET}}$	53	50	83	72
I/O	54	51	84	73
I/O	*	*	85	74
N/C	*	*	*	75
I/O	55	52	86	76
I/O	*	*	87	77
I/O	56	53	88	78

Pin Name	84-Pin PLCC	100-Pin TQFP	132-Pin PQFP	144-Pin TQFP
I/O	*	54	89	79
I/O	57	55	90	80
I/O	*	*	91	81
I/O	58	56	92	82
I/O	*	57	93	83
I/O	59	58	94	84
I/O	*	*	95	85
GND	60	59	96	86
VSS	61	60	97	87
I/O	62	61	98	88
N/C	*	*	*	89
I/O	63	62	99	90
I/O	64	63	100	91
I/O	65	64	101	92
I/O	66	65	102	93
VDD	67	66	103	94
VCC	68	67	104	95
I/O	*	*	105	96
I/O	69	68	106	97
I/O	*	69	107	98
I/O	70	70	108	99
I/O	*	*	109	100
GND	*	*	110	101
I/O	71	71	111	102
I/O	*	72	112	103
I/O	72	73	113	104
I/O	*	*	114	105
N/C	*	*	*	106
I/O	73	74	115	107
M2	74	75	116	108
M1	75	76	117	109
I/O	76	77	118	110
N/C	*	*	*	111
I/O	*	*	119	112
I/O	77	78	120	113
I/O	*	*	121	114
I/O	78	79	122	115
I/O	*	80	123	116
I/O	79	81	124	117
I/O	*	*	125	118
I/O	80	82	126	119
I/O	*	83	127	120
I/O	81	84	128	121
I/O	*	*	129	122
GND	82	85	130	123
I/O	83	86	131	124
N/C	*	*	*	125
A16 or I/O	84	87	132	126

Notes: 1. * Indicates unconnected package pin.

2. Contact Atmel for availability.

A.C. Timing Characteristics

Delays are based on fixed load. Loads for each type of device are described in the notes. Delays are in nanoseconds. Worst case: Vcc = 4.75 V to 5.25 V. Temperature = 0°C to 70°C.

Cell Function	Parameter	From	To	Load	- 2	- 4	Units
Wire (4)	t_{PD} (max) (4)	A, B, L	A, B	1	1.2	1.8	ns
NAND	t_{PD} (max)	A, B, L	B	1	2.2	3.2	ns
XOR	t_{PD} (max)	A, B, L	A	1	2.4	4.0	ns
AND	t_{PD} (max)	A, B, L	B	1	2.2	3.2	ns
MUX	t_{PD} (max)	A, B	A	1	2.3	4.0	ns
		L	A	1	3.0	4.9	ns
D-Flip-Flop (5)	t_{setup} (min)	A, B, L	CLK		2.0	3.0	ns
D-Flip-Flop (5)	t_{hold} (min)	CLK	A, B, L		0.0	0.0	ns
D-Flip-Flop	t_{PD} (max)	CLK	A	1	2.0	3.0	ns
Bus Driver	t_{PD} (max)	A	L	2	2.6	4.0	ns
Repeater	t_{PD} (max)	L, E	E	3	1.6	2.3	ns
		L, E	L	2	2.1	3.0	ns
Column Clock	t_{PD} (max)	GCLK, A, ES	CLK	3	2.4	3.0	ns
Column Reset	t_{PD} (max)	GRES, A, EN	RES	3	2.4	3.0	ns
Clock Buffer (5)	t_{PD} (max)	CLOCK PIN	GCLK	4	2.0	2.9	ns
Reset Buffer (5)	t_{PD} (max)	RESET PIN	GRES	5	1.9	2.8	ns
TTL Input (1)	t_{PD} (max)	I/O	A	3	1.2	1.5	ns
CMOS Input (2)	t_{PD} (max)	I/O	A	3	1.4	2.3	ns
Fast Output (3)	t_{PD} (max)	A	I/O PIN	6	3.5	6.0	ns
Slow Output (3)	t_{PD} (max)	A	I/O PIN	6	8.0	12.0	ns
Output Disable (5)	t_{PXZ} (max)	L	I/O PIN	6	3.3	5.5	ns
Fast Enable (3, 5)	t_{PZX} (max)	L	I/O PIN	6	4.0	6.5	ns
Slow Enable (3, 5)	t_{PZX} (max)	L	I/O PIN	6	8.5	12.5	ns

Device	Cell Types	Outputs	-2 Icc (max)	-4 Icc (max)
Cell (6)	Wire, XWire, Half-Adder, Flip-Flop	A, B	4.5 µA/MHz	7 µA/MHz
Bus (6)	Wire, XWire, Half-Adder, Flip-Flop, Repeater	L	2.5 µA/MHz	4 µA/MHz
Column Clock (6)	Column Clock Driver	CLK	40 µA/MHz	60 µA/MHz

Notes:

1. TTL buffer delays are measured from a V_{IH} of 1.5 V at the pad to the internal V_{IH} at A. The input buffer load is constant.
2. CMOS buffer delays are measured from a V_{IH} of 1/2 V_{CC} at the pad to the internal V_{IH} at A. The input buffer load is constant.
3. Buffer delay is to a pad V_{IH} of 1.5 V with one output switching.
4. Max specifications are the average of max t_{PDLH} and t_{PDHL}.
5. Parameter based on characterization and simulation; not tested in production.
6. Exact power calculation is available in an Atmel application note.

Load Definition:

1. Load of one A or B input
2. Load of one L input
3. Constant Load
4. Load of 28 Clock Columns
5. Load of 28 Reset Columns
6. Tester Load of 50 pF

Absolute Maximum Ratings*

Supply Voltage (V_{CC})	-0.5 V to +7.0 V
DC Input Voltage (V_{IN})	-0.5 V to VCC + 0.5 V
DC Output Voltage (V_{ON})	-0.5 V to VCC + 0.5 V
Storage Temperature Range (T_{STG})	-65°C to +150°C
Power Dissipation (P_D)	1500 mW
Lead Temperature (T_L) (Soldering, 10 sec.)	260°C
ESD (R_{ZAP}=1.5K, C_{ZAP}=100 pF)	2000 V

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

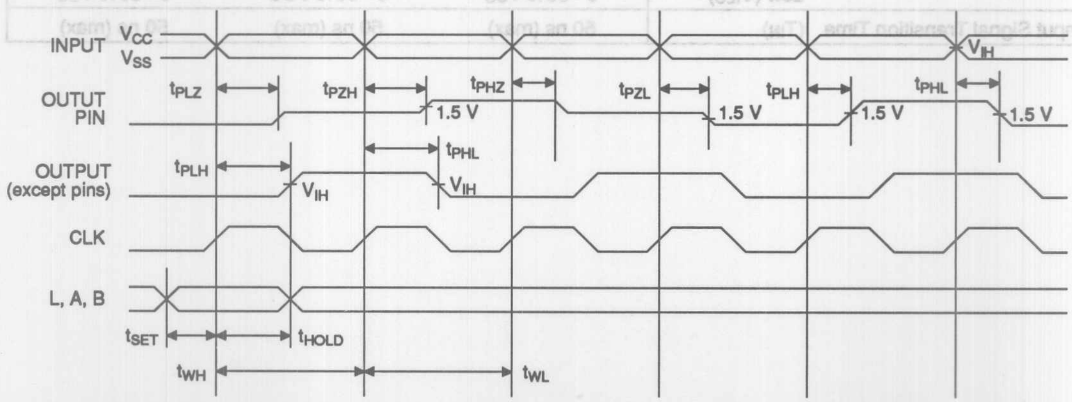
D.C. and A.C. Operating Range

		AT6002-2/4 AT6005-2/4 Com.	AT6002-4 AT6005-4 Ind.	AT6002-4 AT6005-4 Mil.
Operating Temperature (Case)		0°C - 70°C	-40°C - 85°C	-55°C - 125°C
V_{CC} Power Supply		5 V ± 5%	5 V ± 10%	5 V ± 10%
Input Voltage Level (TTL)	High (V_{IHT})	2.0 V - V_{CC}	2.0 V - V_{CC}	2.0 V - V_{CC}
	Low (V_{ILT})	0 V - 0.8 V	0 V - 0.8 V	0 V - 0.8 V
Input Voltage Level (CMOS)	High (V_{IHC})	70% - 100% V_{CC}	70% - 100% V_{CC}	70% - 100% V_{CC}
	Low (V_{ILC})	0 - 30% V_{CC}	0 - 30% VCC	0 - 30% V_{CC}
Input Signal Transition Time	(T_{IN})	50 ns (max)	50 ns (max)	50 ns (max)

D.C. Characteristics

Symbol	Parameter	Conditions		Min	Max	Units
V_{IH}	High-Level Input Voltage	Commercial	CMOS	70% V_{CC}	V_{CC}	V
			TTL	2.0	V_{CC}	V
V_{IL}	Low-Level Input Voltage	Commercial	CMOS	0	20% V_{CC}	V
			TTL	0	0.8	V
V_{OH}	High-Level Output Voltage	Commercial	I_{OH} = -4 mA, V_{CC} min	3.9		V
			I_{OH} = -12 mA, V_{CC} min	3.0		V
V_{OL}	Low-Level Output Voltage	Commercial	I_{OL} = -4 mA, V_{CC} min		0.4	V
			I_{OL} = -12 mA, V_{CC} min		0.5	V
I_{OZH}	High-Level Tristate Output Leakage Current	V_O = V_{CC} (max)			10	µA
I_{OZL}	Low-Level Tristate	Without Pull-Up, V_O = V_{SS}		-10		µA
	Output Leakage Current	With Pull-Up, V_O = V_{SS}		-500		µA
I_{IH}	High-Level Input Current	VIN = V_{CC} (max)			10	µA
I_{IL}	Low-Level Input Current	Without Pull-Up, V_{IN} = V_{SS}		-10		µA
		With Pull-Up, V_{IN} = V_{SS}		-500		µA
I_{CC}	Power Consumption	Without Internal Oscillator (Standby)			500	µA
C_{IN}	Input Capacitance	All Pins			10	pF

Device Timing: During Operation



Cache Logic™ is a trademark of Atmel Corporation.

Ordering Information

Usable Gates	Speed Grade (ns)	Ordering Code	Package	Operation Range
2,000	2	AT6002-2AC AT6002-2JC AT6002-2QC	100A 84J 132Q	Commercial (0°C to 70°C)
2,000	4	AT6002-4AC AT6002-4JC AT6002-4QC	100A 84J 132Q	Commercial (0°C to 70°C)
		AT6002-4AI AT6002-4JI AT6002-4QI	100A 84J 132Q	Industrial (-40°C to 85°C)

Usable Gates	Speed Grade (ns)	Ordering Code	Package	Operation Range
5,000	2	AT6005-2AC AT6005A-2AC AT6005-2JC AT6005-2QC	100A 144A 84J 132Q	Commercial (0°C to 70°C)
5,000	4	AT6005-4AC AT6005A-4AC AT6005-4JC AT6005-4QC	100A 144A 84J 132Q	Commercial (0°C to 70°C)
		AT6005-4AI AT6005A-4AI AT6005-4JI AT6005-4QI	100A 144A 84J 132Q	Industrial (-40°C to 85°C)

Package Type	
100A	100 Lead, Very Thin (1.0 mm) Plastic Gull Wing Quad Flat Package (VQFP)
144A	144 Lead, Thin (1.4 mm) Plastic Gull Wing Quad Flat Package (TQFP)
84J	84 Lead, Plastic J-Leaded Chip Carrier (PLCC)
132Q	132 Lead, Plastic Gull Wing Quad Flat Package (PQFP)

AT6000 Series Configuration

Field Programmable Gate Array

Configuration is the process of loading a design into a AT6000 Series field programmable gate array (FPGA). AT6000 Series devices are SRAM-based, and can be configured any number of times. The entire device or select portions of a design can be configured. Sections of the device can be configured while others continue to operate undisturbed.

Configuration data is transferred to the device in one of seven modes. Full configuration takes only milliseconds. Partial configuration takes even less— it is a function of design density.

Configuration is controlled by dedicated configuration pins and dual-function pins that double as I/O pins when the device is operating. Three pins, M0, M1 and M2 determine the configuration mode (Table 1). The number of dual-function pins required for each mode varies (Table 2).

One of the modes is automatically initiated after power-up reboot; the others are initiated by the user. Configuration data can come from a variety of external logic sources, including a PC parallel port, microprocessor or EPROM.

The user determines the configuration mode for loading the bit pattern into the device. The Integrated Development System generates the SRAM bit pattern required to configure a AT6000 Series FPGA. Many factors can influence the user's choice of configuration mode, including device size, board space, required configuration speed, number of devices to be configured, and design size.

This application note suggests guidelines for device configuration and describes each of the configuration modes in detail.

A basic understanding of the device architecture, as described in the AT6000 Series data sheet, is assumed.

Features

Variety of Formats

- PC Parallel Port
- Microprocessor
- Serial/Parallel EPROM

Configuration Windows

- Full or Partial Reconfiguration
- Bit-Stream Compression Algorithm

Reprogrammable

- Download Configuration any Number of Times
- Reconfigure In-System Down to Cell Level

Fast

- Full Configuration: 1-8 Milliseconds
- Partial Configuration: 0.2 µseconds/Cell

Table 1. AT6000 Series Configuration Modes

Mode	Description	M2	M1	M0	Application
0	Configuration Reset	0	0	0	Clearing the Device
1	Address Count-Up, External CCLK	0	0	1	Fast Configuration; Parallel EPROM
2	Address Count-Down, External CCLK	0	1	0	Fast Configuration; Parallel EPROM
3	Bit-Sequential, External CCLK	0	1	1	Serial Communication Port to UART
4	Bit-Sequential, Internal CCLK	1	0	0	Serial EPROM; Auto Configuration
5	Address Count-Up, Internal CCLK	1	0	1	Parallel EPROM
6	Byte-Sequential, External CCLK	1	1	0	Parallel Port of Microprocessor

Configuration Modes

Powering up a AT6000 Series FPGA is a three-step process. When power is first applied, the device enters an initialization state that takes about 8 milliseconds and resets the SRAM to all zeros. Cells in the array become cross wires with no A or B inputs selected, all bus drivers are switched off, repeaters are disabled, I/Os are set as TTL inputs with the pull-up enabled, column clocks are set to "0," and column resets are set to "1."

After initialization, the device enters the configuration state and writes to the memory bits that control cell functionality and interconnection.

Seven configuration modes are available:

Mode 0:	Configuration Reset
Mode 1:	Address Count-Up, External CCLK
Mode 2:	Address Count-Down, External CCLK
Mode 3:	Bit-Sequential, External CCLK
Mode 4:	Bit Sequential, Internal CCLK
Mode 5:	Address Count-Up, Internal CCLK
Mode 6:	Byte-Sequential, External CCLK

Mode 0 is not a true configuration mode because it does not load a design into the FPGA. Instead, mode 0 initiates the reboot sequence and clears the device, preparing it for configuration or reconfiguration.

Modes 1, 2 and 5 generate external address outputs so the user can conveniently access sequential data from a standard parallel EPROM. The generated output addresses bear no relation to the internal addresses of the FPGA's configuration SRAM, they simply count up or down with each CCLK edge to create a sequential byte stream. Mode 6 is similar to modes 1, 2 and 5 but assumes a system-generated bit stream and does not generate external address outputs. Modes 3 and 4 use a serial bit stream received from the system, an industry-standard EPROM or the download cable provided with the Integrated Development System. The data in each byte is serialized with the least-significant-bit supplied first. Mode 4 can be initiated automatically by the FPGA.

Modes 3 and 4 will typically be the most popular configuration modes because they require the fewest pins and receive data from small foot-print serial EPROMs that take up little board space.

Pins used for Configuration

AT6000 Series FPGAs have three kinds of pins: dedicated I/O pins, dedicated configuration pins, and dual-function pins which act as I/O during operation but are used for various control signals during configuration. (For more on device pins refer to the AT6000 Series data sheet.)

Dedicated Configuration Pins

There are six signals dedicated to programming: M0, M1, M2, CCLK, $\overline{\text{CON}}$ and $\overline{\text{CS}}$.

M0, M1, M2

The mode pins are inputs that determine the configuration mode to be used. Table 1 (front page) lists the states for each configuration mode. M0, M1 and M2 can be fixed in modes 1 through 6 and ignored. Mode 0, configuration reset, can be initiated by asynchronously driving M0, M1 and M2 low, then returning them to the proper mode selection value.

CCLK

CCLK is the configuration clock signal. It is an input or an output depending on the mode of operation. In modes 1, 2, 3 and 6 it is an input, in modes 4 and 5 it is an output with a typical frequency of 1 MHz. In all modes, the rising edge of the CCLK signal is used to sample inputs and change outputs.

Table 2. Dual-Function Pin Usage

N = Not used, R = Required, O = Optional

Mode	Minimum Dual-Function Pins	Optional Dual-Function Pins	A0-16 Outputs	D0 Input	D1-7 Inputs	$\overline{CEN}$ Output	$\overline{CHECK}$ Output	$\overline{ERR}$ Output	$\overline{CSOUT}$ Output
0	0	0	N	N	N	N	N	N	N
1	25	4	R	R	R	O	O	O	O
2	25	4	R	R	R	O	O	O	O
3	1	3	N	R	N	N	O	O	O
4	1	3	N	R	N	N	O	O	O
5	25	4	R	R	R	O	O	O	O
6	8	3	N	R	R	N	O	O	O

$\overline{CON}$

$\overline{CON}$ is a bidirectional open-collector pin that provides the configuration control and status signal. Configuration starts on the first CCLK edge when $\overline{CON}$ is driven and held low. Configuration continues until $\overline{CON}$ is pulled high by a pull-up or by the configuration system. $\overline{CON}$ is driven low by the device until configuration is complete. The device moves to the operation state on the first CCLK edge after $\overline{CON}$ is high.

$\overline{CS}$

$\overline{CS}$ is the configuration chip select pin. $\overline{CS}$ must be low for configuration to occur. Pulling $\overline{CS}$ high during configuration does not stop the process, but the pin should be held low throughout configuration. $\overline{CS}$ can be used to cascade devices (see Figures 9 and 14) and create an addressed, multiple-device programming system (see Figure 15).

Dual-Function Pins

Dual-function pins are programming pins during configuration and I/O pins during operation. The number of dual-function pins used during configuration varies from mode to mode. Some dual-function pins act as configuration status pins and are optional regardless of mode. The optional pins are most useful when cascading devices to program multiple FPGAs from a single data source. Table 2 lists the dual-function pins used with each mode.

D0-D7

D0-D7 are data input pins. Parallel modes 1, 2, 5 and 6 use all eight data inputs, serial modes 3 and 4 use only one, D0.

A0-A16

A0-A16 are address output signals, used by modes 1, 2 and 5, to drive an EPROM or other external addressed-memory device.

$\overline{CSOUT}$

$\overline{CSOUT}$ drives $\overline{CS}$ of the next device in a configuration chain.

$\overline{CHECK}$

$\overline{CHECK}$ is an input that enables an internal SRAM checking feature.

$\overline{ERR}$

$\overline{ERR}$ is an output that switches low when an error is detected. It is used with the $\overline{CHECK}$ function, or when protocol errors occur during configuration.

$\overline{CHECK}$ and $\overline{ERR}$ work together to perform simple and advanced diagnostic tests. For example, they can be used to verify the accuracy of a configuration run. With the $\overline{CHECK}$ pin low, download the configuration file a second time. The device systematically compares the data values in the configuration file with the data already programmed into the device's SRAM. If a mismatch is found the $\overline{ERR}$ pin switches and remains low until the end of the configuration cycle.

Pin Status

The status of dual-function pins is determined by the device state. All I/Os are disabled during initialization. To move from the initialization state to configuration, the $\overline{CON}$ and $\overline{CS}$ pins are driven low. During configuration, the dual-function pins used by the selected mode are converted to inputs and outputs as required. To move from the configuration state to operation, the configuration file must be loaded completely and either $\overline{CON}$ or $\overline{CS}$ must be high. During operation, the I/O pins behave according to the specified design.

Control Register

The Integrated Development System generates the bit-stream file used to configure the FPGA. In addition to the actual data to be loaded into the SRAM, the bit stream loads a control register containing eight bits used to control various configuration sequence parameters (Figure 1).

Figure 1. Control Register

B7	B6	B5	B4	B3	B2	B1	B0

B0

B0 controls the value of the device's memory address counter after each configuration sequence. The default resets the value, so subsequent configuration sequences load the configuration file from the same address. In modes 1 and 5, the default address is 0000, in mode 2 it is 1FFFF. When B0 is set, the memory address counter retains its last value, so the user can store multiple designs sequentially in an EPROM.

B1

B1 controls loading of a jump address into the device's memory address counter. The default ignores any jump addresses. With B1 set, the memory address counter jumps to the specified address. Using B1, configuration files can be stored as a continuous stream or as a pointer-based list.

B2

B2 controls operation of the dual-function pins $\overline{\text{CSOUT}}$ and $\overline{\text{CEN}}$. When B2 is set, these two pins are disabled. This is useful when a minimum pin-count configuration is desired.

B3

B3 controls the operation of the dual-function pins $\overline{\text{ERR}}$ and $\overline{\text{CHECK}}$. When B3 is set, both pins are disabled. This is useful when a minimum pin-count configuration is desired, or when design security is a concern.

B4

B4 controls the writing of configuration data after the initialization state. When B4 is set, configuration data can not be written into the device by subsequent configuration cycles. B4 can only be reset by rebooting the device.

B5 and B6

B5 and B6 control the operation of the global clock signal received through the CLOCK pin:

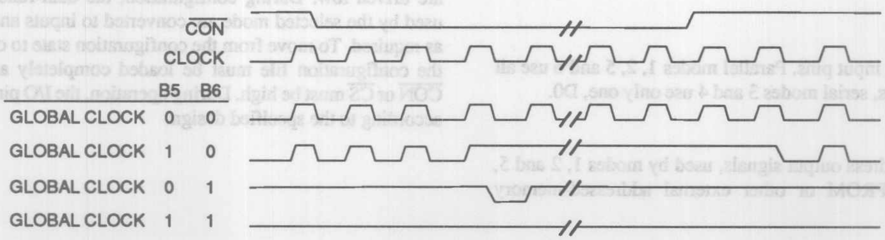
B5	B6	Global Clock Operation
0	0	Normal operation.
1	0	Stops after third rising edge of CLOCK after $\overline{\text{CON}}$ is low. Continues after second rising edge of CLOCK after $\overline{\text{CON}}$ is high.
0	1	Stops after fourth rising edge of CLOCK after $\overline{\text{CON}}$ is high. Each configuration cycle thereafter, it receives one pulse after the third rising edge of CLOCK after $\overline{\text{CON}}$ is low.
1	1	Stops at second rising edge of CLOCK after $\overline{\text{CON}}$ is high. Remains stopped regardless of $\overline{\text{CON}}$.

Figure 2 shows the waveforms associated with each combination. B5 and B6 are not available on the AT6005 device.

B7

B7 controls static power consumption. When B7 is set, the internal oscillator used for auto configuration in modes 4 and 5 is disabled. On the AT6005, this function is performed by setting B4.

Figure 2. Global Clock Signal During Operation



Configuration State Machine

Configuration is executed by a synchronous state machine that controls the flow of configuration data into the FPGA (Figure 3). The state machine is clocked by CCLK whether the signal is externally supplied or generated internally. On each CCLK cycle a different byte or bit of the configuration file is loaded into the state machine.

Data flow is controlled by the external input signals M0, M1, M2, $\overline{\text{CON}}$, $\overline{\text{CS}}$, $\overline{\text{CHECK}}$, D0-D7 and the values in the configuration control register. The state machine generates all the internal control signals as well as the A0-A16 output signals, $\overline{\text{ERR}}$ output, and $\overline{\text{CSOUT}}$ signal. Data is loaded into the device in a stream format and has no absolute address.

The process starts on power-up or when a mode 0 reset is applied. The reboot phase lasts for approximately 8000 internal clock cycles while all the internal SRAM cells are written to a "0" value. During reboot the mode pins are sampled and the configuration-clock output starts.

Mode 4 supports automatic configuration. During reboot, the CCLK pin is enabled for output. After reboot, mode 4 releases the $\overline{\text{CON}}$ pin, allowing it to be pulled high, and then drives it low again to begin a configuration cycle automatically.

In the modes 1,2,3 and 6, CCLK remains an input but is ignored until the reboot process is complete. After reboot, the other modes release $\overline{\text{CON}}$ and allow it to float high. Control of the state machine is then transferred from the internal clock to the CCLK input signal. The device remains in idle until the $\overline{\text{CON}}$ pin is driven low.

Driving $\overline{\text{CON}}$ low puts the device in the preamble check loop, a synchronizing procedure for both parallel and serial configuration modes. Configuration is dependent on the sequence of data, and the preamble specifies the first byte of the data stream. In modes 3 and 4, the preamble also defines the byte boundary of serial data, which may be parallel before being processed by the state machine.

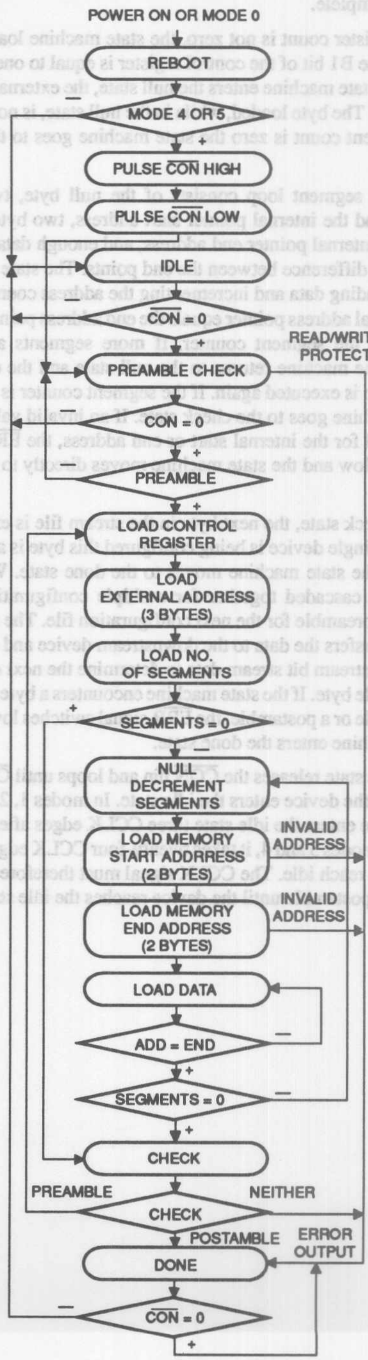
The first error test is done during the preamble check. If a read or write protect bit appears in the control register of the preamble, the $\overline{\text{ERR}}$ pin goes low and the device goes directly to the done state.

After the preamble check, the state machine loads the configuration control register and drives and holds the $\overline{\text{CON}}$ signal low until configuration is complete. Loading the configuration control register puts the next byte in the stream file into the control register, which controls features and variations in the configuration process.

The state machine then loads the next three bytes into a temporary register used to parallel-load the external address counter. This action is similar to a microprocessor "jump" command. The address bytes are loaded in all modes, but the jump can only be used by modes 1, 2 and 5.

The next byte in the stream file indicates the number of data strings, or segments. A single file can have 0 to 255 segments. The byte is loaded into a counter which is decremented at the

Figure 3. Configuration State Machine



end of each data string until the it reaches zero and configuration is complete.

If the register count is not zero, the state machine loads a null byte. If the B1 bit of the control register is equal to one the first time the state machine enters the null state, the external address is loaded. The byte loaded, while in the null state, is not used. If the segment count is zero the state machine goes to the check state.

The data segment loop consists of the null byte, two bytes which load the internal pointer start address, two bytes which load the internal pointer end address, and enough data bytes to equal the difference between the end points. The state machine loops, loading data and incrementing the address counter, until the internal address pointer equals the end address pointer. Then it checks the segment counter. If more segments are to be loaded, the machine returns to the null state and the data segment loop is executed again. If the segment counter is zero, the state machine goes to the check state. If an invalid value is encountered for the internal start or end address, the $\overline{\text{ERR}}$ signal switches low and the state machine moves directly to the done state.

In the check state, the next byte in the stream file is examined. When a single device is being configured this byte is a postamble and the state machine moves to the done state. When devices are cascaded together for multiple configurations, this byte is a preamble for the next configuration file. The state machine transfers the data to the downstream device and monitors the downstream bit stream data to determine the next check for a preamble byte. If the state machine encounters a byte that isn't a preamble or a postamble, the $\overline{\text{ERR}}$ signal switches low and the state machine enters the done state.

The done state releases the $\overline{\text{CON}}$ pin and loops until $\overline{\text{CON}}$ goes high and the device enters the idle state. In modes 1, 2, 5 and 6, the device enters the idle state three CCLK edges after $\overline{\text{CON}}$ is high. In modes 3 and 4, it takes twenty-four CCLK edges for the device to reach idle. The CCLK signal must therefore continue after the postamble until the device reaches the idle state.

Partial Configuration

Figure 4 gives the bit stream file used to configure a hypothetical device that has a 6 x 6 array of cells lined with four I/O on each side. Configuration begins with the bottom left cell, number 0, and ends with the upper right cell, number 35. Then the I/O cells are configured, beginning at number 36 and proceeding clockwise to number 51.

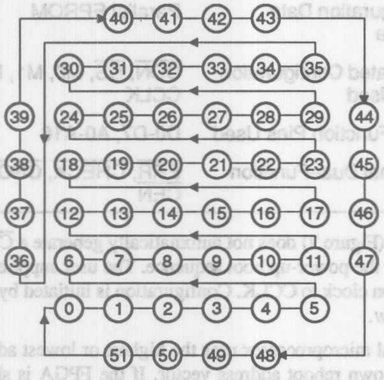
By placing windows in the bit stream file, it is possible to configure only a portion of the array. Figure 5 gives the bit stream file used to configure the lower right portion of the array and I/Os 45, 46 and 47— the program leaves darkened cells untouched.

On line six the program states that there are four segments of data to configure. The start address is the left-most cell in the bottom row to be configured, number 9, and the end address is the right-most cell in the row, number 11. Four bytes of data are used to load the cell between the two points, number 10. The next segment configures the row above. Notice that the cells between the first segment end address and the second segment start address are omitted. The data in these cells is left untouched— only the cell being programmed on a given clock cycle is changed. The other cells function as if in their normal operational mode. This means a portion of the array can be configured while the rest of the array remains operational.

Configuration Compression

A configuration compression algorithm, included in the Integrated Development System, uses windowing to compress the configuration file. On power-up, all cells in the FPGA are programmed to be logical zeros. Unused cells in a design remain zeros, so they do not need to be configured. The compression algorithm skips unused cells and can reduce file size by up tp 80%. This in turn reduces configuration time and memory storage requirements. It even makes designs less susceptible to reverse engineering, due to the random start and end array addresses in the compressed bit stream. For more information about the configuration compression algorithm, refer to the FPGA application notes.

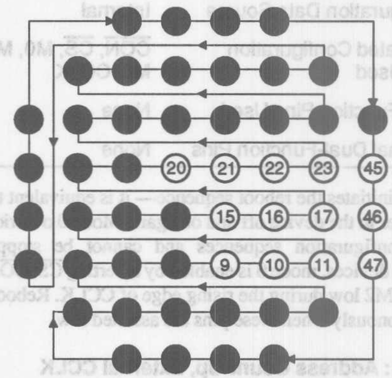
Figure 4. Full Configuration Example



Full Configuration

```
10110010  ; preamble
00000000  ; control register
00000000  ; external address msb
00000000  ;
00000000  ; external address lsb
11111110  ; number of window segments
00000000  ; null
00000000  ; start address
00000000  ;
00000000  ; end address
00110011  ;
00000000  ; data
00000001
00000010
00000011
00000100
.
.
00110010
00110011
01001101  ; postamble
```

Figure 5. Partial Configuration Example



Partial Configuration

```
10110010  ; preamble
00000000  ; control register
00000000  ; external address msb
00000000  ;
00000000  ; external address lsb
11111011  ; number of window segments
00000000  ; null
00000000  ; start address segment 1
00001001  ;
00000000  ; end address
00010011  ;
00001001  ; data
00001010
00001011
00000000  ; null
00000000  ; start address segment 2
00001111  ;
00000000  ; end address
00010001  ;
00001111  ; data
00010001
00010001
.
.
00000000  ; null
00000000  ; start address segment 4
00101101  ;
00000000  ; end address
00101111  ;
00101101  ; data
00101110
00101111
01001101  ; postamble
```

Configuration Modes

This section gives setup requirements and usage guidelines for each configuration mode.

Mode 0: Configuration Reset

Configuration Data Source	Internal
Dedicated Configuration Pins Used	$\overline{CON}$, $\overline{CS}$, M0, M1, M2, CCLK
Dual-Function Pins Used	None
Optional Dual-Function Pins	None

Mode 0 initiates the reboot sequence— it is equivalent to turning power to the device off and on again. Mode 0 overrides any other configuration sequences and cannot be stopped. In AT6000 devices, mode 0 is enabled by asserting $\overline{CS}$, $\overline{CON}$, M0, M1 and M2 low during the rising edge of CCLK. Reboot starts asynchronously when these pins are asserted low.

Mode 1: Address Count-Up, External CCLK

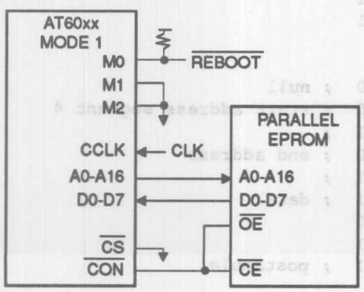
Configuration Data Source	Parallel EPROM
Dedicated Configuration Pins Used	$\overline{CON}$, $\overline{CS}$, M0, M1, M2, CCLK
Dual-Function Pins Used	D0-D7, A0-A16
Optional Dual-Function Pins	$\overline{ERR}$, $\overline{CHECK}$, $\overline{CSOUT}$, $\overline{CEN}$

Mode 1 (Figure 6) does not automatically generate a $\overline{CON}$ signal after the power-up boot sequence. The user supplies a configuration clock to CCLK. Configuration is initiated by driving $\overline{CON}$ low.

If the FPGA is sharing a large EPROM with a microprocessor, the addresses used must start from the opposite end of the EPROM address map so that configuration does not interfere with the microprocessor, and vice versa. In mode 1, the external address counter starts at 00000 and counts up (see mode 2 description).

Using a maximum clock rate of 10 MHz, mode 1 can configure a single device in under 1 millisecond. Cascading devices limits the parallel data rate to 800 kHz.

Figure 6. Mode 1 Configuration



Mode 2: Address Count-Down, External CCLK

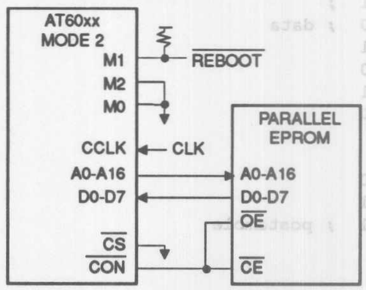
Configuration Data Source	Parallel EPROM
Dedicated Configuration Pins Used	$\overline{CON}$, $\overline{CS}$, M0, M1, M2, CCLK
Dual-Function Pins Used	D0-D7, A0-A16
Optional Dual-Function Pins	$\overline{ERR}$, $\overline{CHECK}$, $\overline{CSOUT}$, $\overline{CEN}$

Mode 2 (Figure 7) does not automatically generate a $\overline{CON}$ signal after the power-up boot sequence. The user supplies a configuration clock to CCLK. Configuration is initiated by driving $\overline{CON}$ low.

A typical microprocessor uses the highest or lowest address to load its own reboot address vector. If the FPGA is sharing a large EPROM with a microprocessor, the addresses used must start from the opposite end of the EPROM address map so that configuration does not interfere with the microprocessor, and vice versa. In mode 2, the external address counter starts at 1FFFF and counts down (see mode 1 description).

Using a maximum clock rate of 10 MHz, mode 2 can configure a single device in under one millisecond. Cascading devices limits the parallel data rate to 800 kHz.

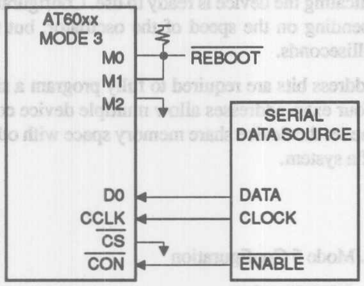
Figure 7. Mode 2 Configuration



Mode 3: Bit-Sequential, External CCLK

Configuration Data Source	Serial EPROM, Serial Comm. Port, UART, Download Cable
Dedicated Configuration Pins Used	$\overline{\text{CON}}$, $\overline{\text{CS}}$, M0, M1, M2, CCLK
Dual-Function Pins Used	D0
Optional Dual-Function Pins	$\overline{\text{ERR}}$, $\overline{\text{CHECK}}$, $\overline{\text{CSOUT}}$

Figure 8. Mode 3 Configuration



Mode 3 (Figure 8) does not automatically generate a $\overline{\text{CON}}$ signal after the power-up boot sequence. The user must supply a configuration clock to CCLK. Configuration is initiated by driving $\overline{\text{CON}}$ low.

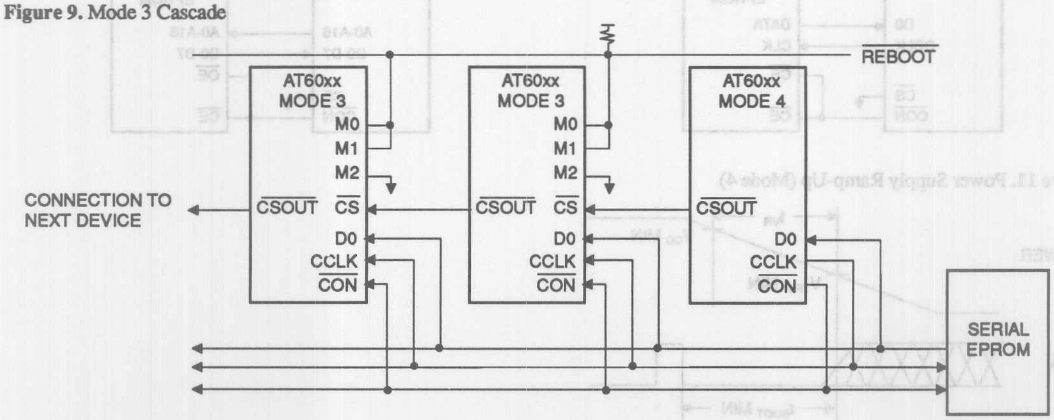
As long as data setup and hold requirements are satisfied, CCLK pulses can have arbitrary periods. This is helpful when using asynchronous communication ports or UARTs for configuration. If CCLK is stopped entirely between configurations, allow 24 preceding and trailing clock pulses with respect to $\overline{\text{CON}}$ going low or high (refer to the AC timing table in the AT6000 Series data sheet).

Depending on the speed of the user-supplied clock, mode 3 configuration can take as little as 8 milliseconds.

Mode 3 can be used to configure multiple devices cascaded together (Figure 9). The first device in the cascade chain must use either Mode 3 or Mode 4. If the configuration file contains a second preamble instead of a postamble (see the configuration-file format section), then the first device in the chain drives $\overline{\text{CSOUT}}$ low enabling the next device in the chain to receive configuration data from the serial data source. Configuration for downstream devices proceeds in a similar manner with "chip select" ($\overline{\text{CS}}$) propagating through the chain.

Mode 3 is used when configuring with the download cable provided in the Integrated Development System.

Figure 9. Mode 3 Cascade



Mode 4: Bit-Sequential, Internal CCLK

Configuration Data Source	Serial EPROM
Dedicated Configuration Pins Used	$\overline{\text{CON}}$, $\overline{\text{CS}}$, M0, M1, M2, CCLK
Dual-Function Pins Used	D0
Optional Dual-Function Pins	$\overline{\text{ERR}}$, $\overline{\text{CHECK}}$, $\overline{\text{CSOUT}}$

Mode 4 (Figure 10) asserts the $\overline{\text{CON}}$ pin low during the power-up boot sequence. $\overline{\text{CON}}$ is released for one CCLK period after initialization to reset the serial EPROM. $\overline{\text{CON}}$ is then automatically re-asserted low and an internal oscillator toggles CCLK. This causes the EPROM to begin downloading configuration data. One bit of data is loaded from the D0 pin on each rising edge of CCLK until configuration is complete. $\overline{\text{CON}}$ is then released, indicating the device is ready to use. Configuration time varies depending on the speed of the internal oscillator, but typically takes about 80 milliseconds.

The power supply ramp-up rate is critical in mode 4. The device generates a reset pulse 8 milliseconds after the supply voltage crosses the V_{TRIP} level (Figure 11). The supply voltage must be at the minimum for the serial EPROM before the FPGA generates its reset pulse. Otherwise, the EPROM's operation may be sporadic.

Figure 10. Mode 4 Configuration

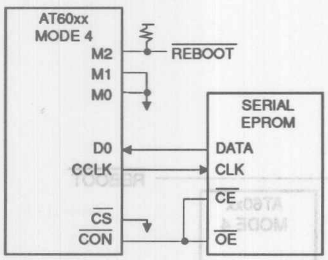
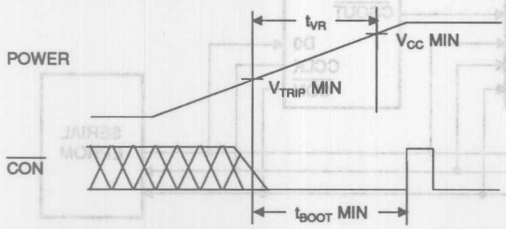


Figure 11. Power Supply Ramp-Up (Mode 4)



V_{CC} min	Minimum voltage for EPROM operation
V_{TRIP} min	Minimum FPGA supply voltage to initiate reboot
t_{BOOT} min	Minimum reboot cycle time
t_{VR}	Minimum rise time of power supply from V_{TRIP} to V_{CC} min.

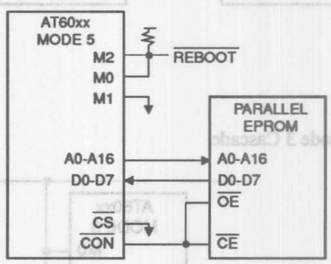
Mode 5: Address Count-Up, Internal CCLK

Configuration Data Source	Parallel EPROM
Dedicated Configuration Pins Used	$\overline{\text{CON}}$, $\overline{\text{CS}}$, M0, M1, M2, CCLK
Dual-Function Pins Used	D0-D7, A0-A16
Optional Dual-Function Pins	$\overline{\text{ERR}}$, $\overline{\text{CHECK}}$, $\overline{\text{CSOUT}}$, $\overline{\text{CEN}}$

Mode 5 (Figure 12) does not automatically generate a $\overline{\text{CON}}$ signal after the power-up boot sequence. Configuration is initiated by driving $\overline{\text{CON}}$ low. An internal oscillator toggles CCLK. This causes the FPGA to generate addresses A0-A16, beginning at 0, to read a configuration file from a parallel EPROM. One byte of configuration data is loaded from the D0-D7 pins on each rising edge of CCLK until configuration is complete. $\overline{\text{CON}}$ is then released, indicating the device is ready to use. Configuration time varies depending on the speed of the oscillator, but typically takes 8 milliseconds.

Only 13 address bits are required to fully program a single device; the four extra addresses allow multiple device configuration and the let the device share memory space with other components of a system.

Figure 12. Mode 5 Configuration



Mode 6: Byte-Sequential, External CCLK

Configuration Data Source	Parallel port of microprocessor
Dedicated Configuration Pins Used	$\overline{CON}$, $\overline{CS}$, M0, M1, M2, CCLK
Dual-Function Pins Used	D0-D7
Optional Dual-Function Pins	$\overline{ERR}$, $\overline{CHECK}$, $\overline{CSOUT}$

Figure 13. Mode 6 Configuration

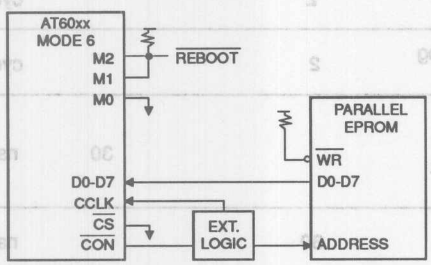


Figure 14. Mode 6 Cascade

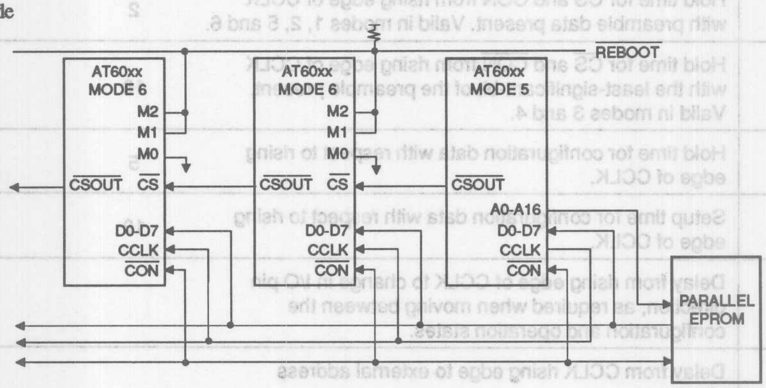
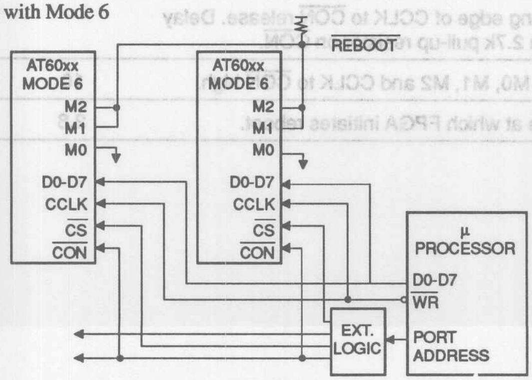


Figure 15. Parallel Configuration with Mode 6



Mode 6 (Figure 13) loads data in 8-bit words to decrease configuration time. It does not use the address pins; each byte in the data stream is setup and held with respect to the rising edge of CCLK.

Mode 6 may be used in a configuration chain (Figure 14) or with the parallel port of a microprocessor or system bus, and may be best for a "smart system" in which the user intends to reconfigure the FPGA as a regular part of system operation. More than one device can be configured by tying all the data buses together and connecting the $\overline{CON}$ pins. The $\overline{CS}$ pin can then be used to select individual devices for configuration (Figure 15).

Configuration Timing Parameters

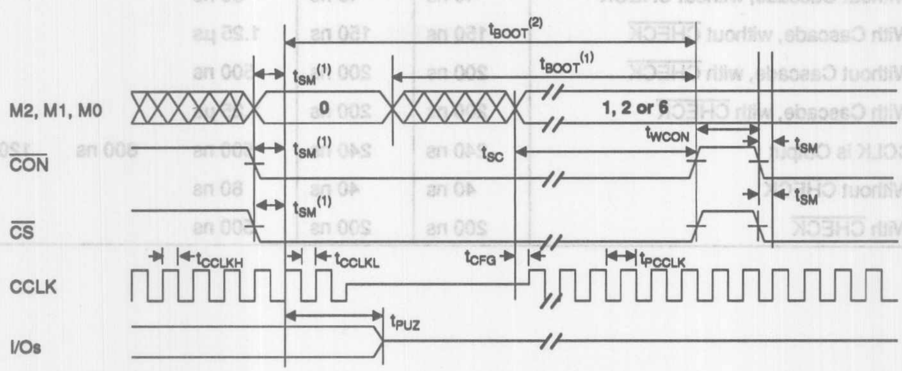
These AC parameters are based on the timing diagrams that follow.

Parameter	Description		Min	Typ	Max	Units
t_{BOOT}	Delay from entry into mode 0 or Power on (V_{CC} > Vsth min) to $\overline{CON}$ released.	AT6002: modes 1,2,3,5,6	1.3	2.0	3.5	ms
		AT6002: mode 4	8	13.0	21.60	ms
		AT6005	4	6.5	10.5	ms
t_{WCON}	$\overline{CON}$ high pulse width. Measured in CCLK clock cycles in modes 1, 2, 3 and 6.		2			cyc
t_{PCON}	CON high pulse width. Measured in CCLK clock cycles in modes 4 and 5.		2			cyc
t_{PUZ}	Delay from power-up or mode 0 to unused I/Os being tri-stated. Measured from the rising edge of CCLK.		2			cyc
t_{DERR}	Delay time from CCLK to change in $\overline{ERR}$. $\overline{ERR}$ will typically be high, and only go low if there is an error during confiugration or a mismatch during the check function.				30	ns
t_{SM}	Setup time from M0, M1, M2, $\overline{CS}$ and $\overline{CON}$ to rising edge of CCLK to initiate configuration or reboot.		30			ns
t_{HMP}	Hold time for $\overline{CS}$ and $\overline{CON}$ from rising edge of CCLK with preamble data present. Valid in modes 1, 2, 5 and 6.		2			cyc
t_{HMS}	Hold time for $\overline{CS}$ and $\overline{CON}$ from rising edge of CCLK with the least-significant-bit of the preamble present. Valid in modes 3 and 4.		16			cyc
t_{HCD}	Hold time for configuration data with respect to rising edge of CCLK.		5			ns
t_{SCD}	Setup time for configuration data with respect to rising edge of CCLK.		10			ns
t_{CFG}	Delay from rising edge of CCLK to change in I/O pin direction, as required when moving between the configuration and operation states.				50	ns
t_{DA}	Delay from CCLK rising edge to external address change.				35	ns
t_{CONH}	Delay from rising edge of CCLK to $\overline{CON}$ release. Delay measured with 2.7k pull-up resistor on $\overline{CON}$.				35	ns
t_{SC}	Setup time for M0, M1, M2 and CCLK to $\overline{CON}$ high.		10			cyc
V_{TRIP}	Supply voltage at which FPGA initiates reboot.		2.8		4.75	V

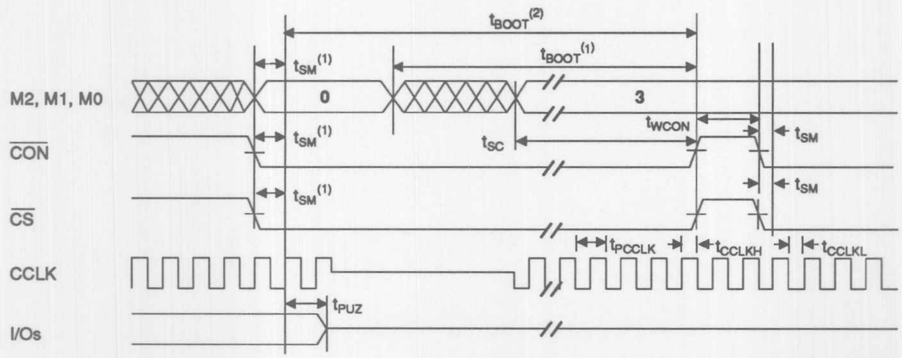
Table 3. CCLK Parameters

Mode(s)	Condition	t_{CCLKL} Min	t_{CCLKH} Min	t_{PCCLK} Min	t_{PCCLK} Typ	t_{PCCLK} Max
1, 2, 6	Without Cascade, without $\overline{\text{CHECK}}$	40 ns	40 ns	80 ns		
1, 2, 6	With Cascade, without $\overline{\text{CHECK}}$	150 ns	150 ns	1.25 µs		
1, 2, 6	Without Cascade, with $\overline{\text{CHECK}}$	200 ns	200 ns	500 ns		
1, 2, 6	With Cascade, with $\overline{\text{CHECK}}$	200 ns	200 ns	1.25 µs		
4, 5	CCLK is Output	240 ns	240 ns	500 ns	800 ns	1200 ns
3	Without $\overline{\text{CHECK}}$	40 ns	40 ns	80 ns		
3	With $\overline{\text{CHECK}}$	200 ns	200 ns	500 ns		

Modes 1, 2 and 6

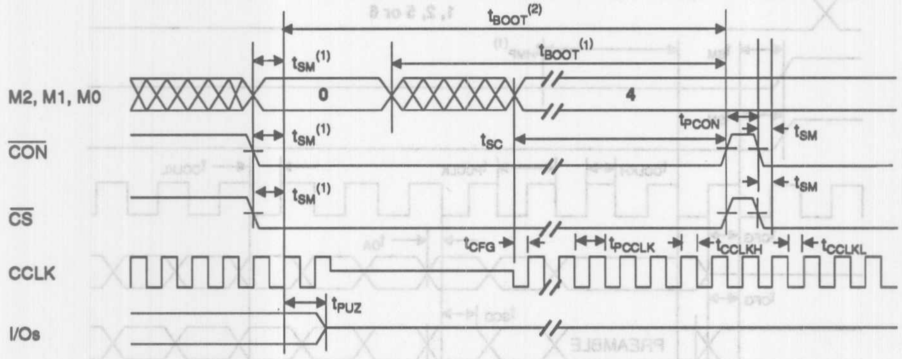


Mode 3



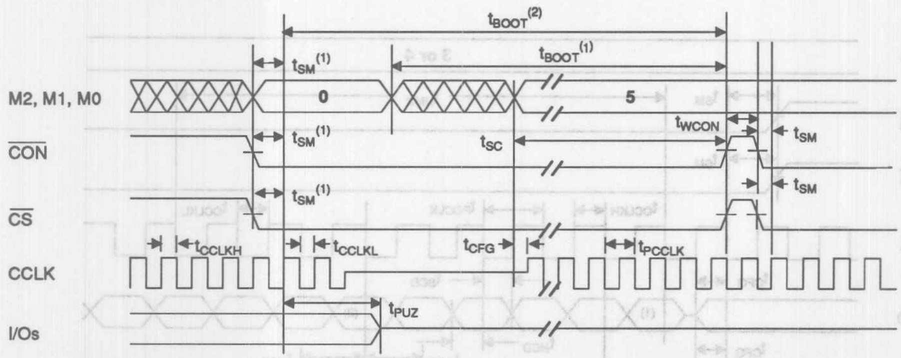
Reboot Cycle (Continued)

Mode 4



Notes: 1. Occurs asynchronously in the AT6002 device.
2. AT6005 device.

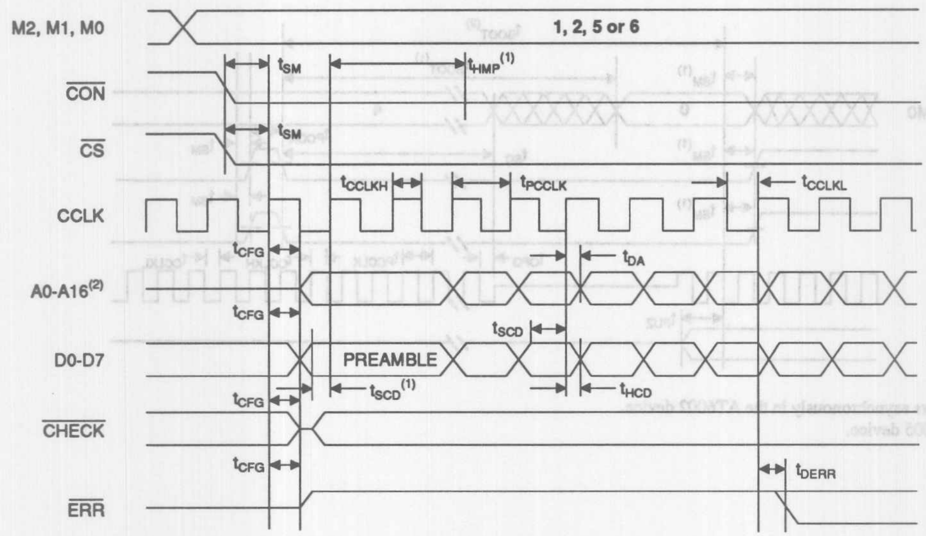
Mode 5



Notes: 1. Occurs asynchronously in the AT6002 device.
2. AT6005 device.

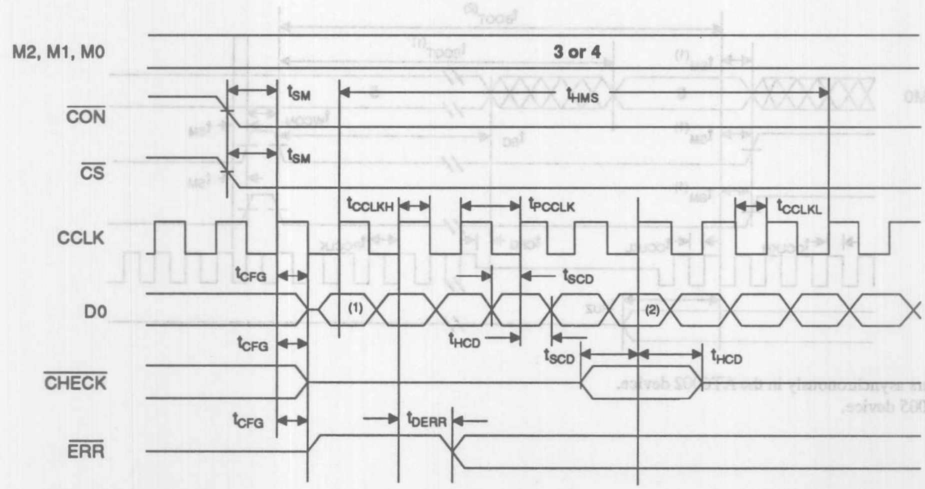
Beginning of Configuration

Modes 1, 2, 5 and 6



Notes: 1. Measured with respect to the edge of CCLK, which clocks in the preamble.
2. A0-A16 not used in mode 6.

Modes 3 and 4

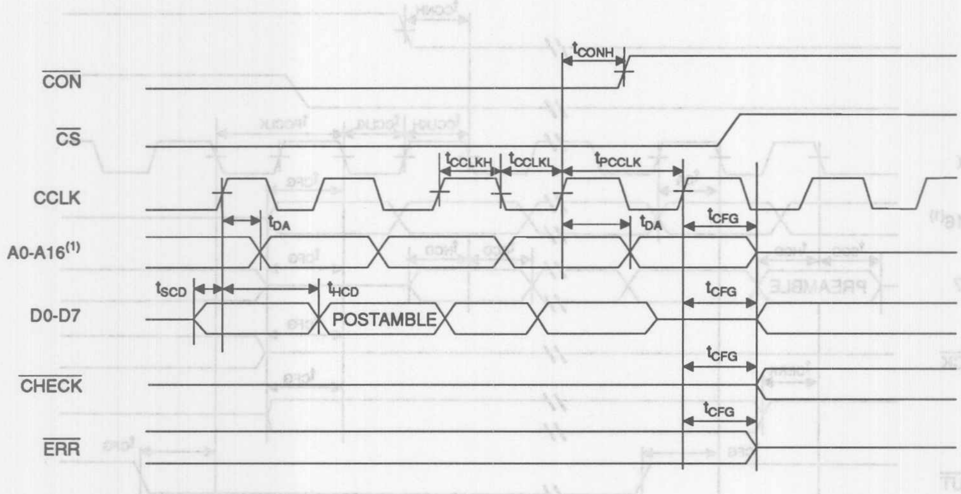


Notes: 1. Preamble LSB.
2. Preamble MSB.

2

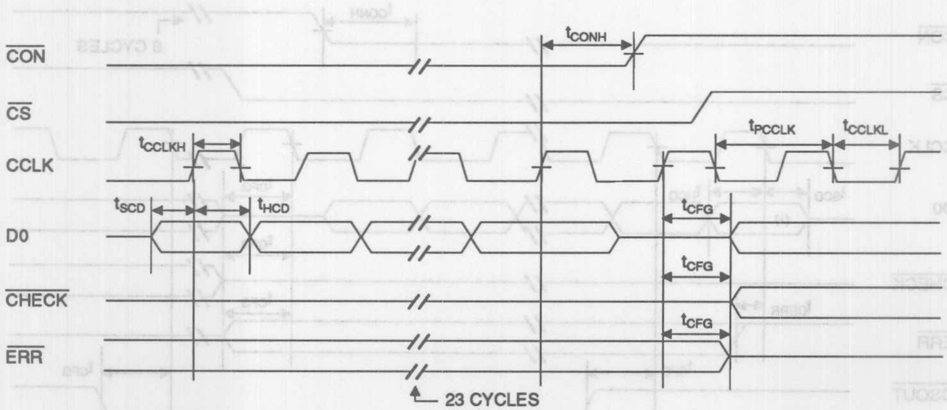
End of Configuration without Cascading

Modes 1, 2, 5 and 6



Note: 1. A0-A16 not used in mode 6.

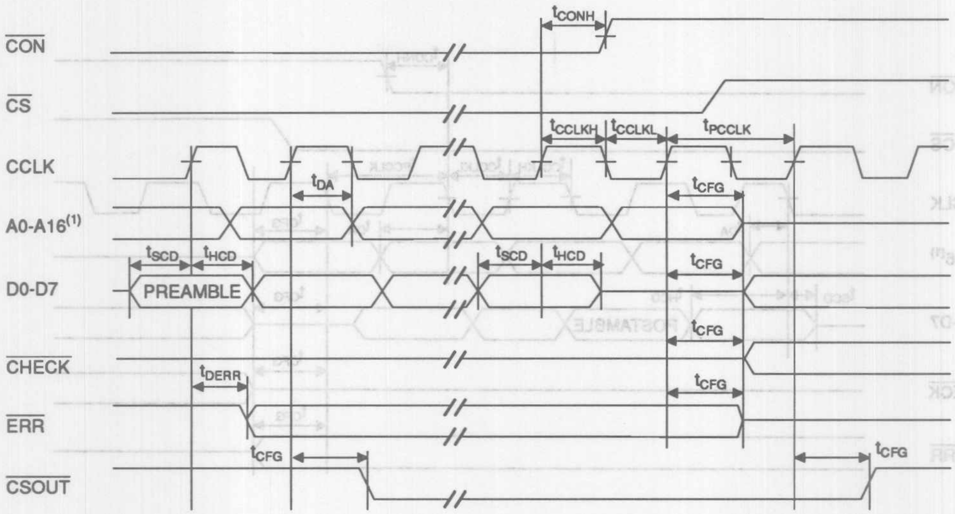
Modes 3 and 4



Note: 1. Postamble LSB.

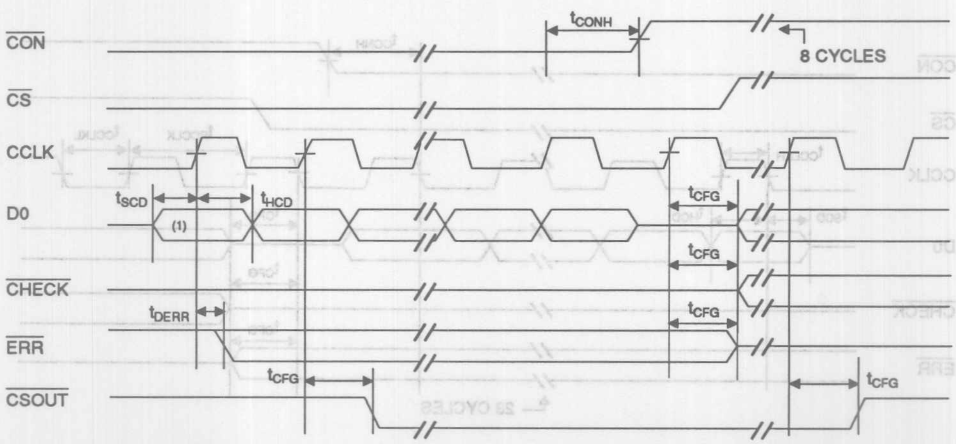
End of Configuration with Cascading

Modes 1, 2, 5 and 6



Note: 1. A0-A16 not used in mode 6.

Modes 3 and 4



Note: 1. Preamble LSB.

Section 3 Programmable Logic Development Tools

CMOS PLD Development Software Support Overview

PLD Software Tools Overview

Atmel's philosophy is that you should be able to use standard tools you already have to design with our programmable logic devices. For those users that do not currently have such a tool, or who wish to augment tools they already own, we have Atmel-specific versions of standard tools available at very reasonable costs.

With the tools Atmel has available, we can serve the needs of beginning users as well as more experienced users. Based on the background of the user, we can make recommendations on the most cost effective solution (see Table 1). If you have any questions regarding which package is best suited to a particular set of needs, please contact the Atmel PLD Technical Support Hotline at (408) 436-4333.

General

Atmel offers versions of standard third-party PLD design tools that are limited to only Atmel PLD devices at a reduced cost compared to the normal versions. These systems are based on software licensed from Data I/O Corporation and Viewlogic Systems, Inc.

Under the terms of our agreements with Data I/O and Viewlogic Systems, Atmel may sell these systems and their options only to users of these limited systems. If a customer already has the full function version from Data I/O and/or Viewlogic Systems, they can purchase upgrades or options from the original company, not from Atmel.

This document outlines general system configurations required to use the various systems, details each package including ordering information, and includes a section with suggested systems for various types of users.

Required System Configurations

At the present time, all of the PLD systems described are available only on PC386/486 platforms running DOS with the system requirements indicated in Table 2. Contact the Atmel PLD Hotline at (408) 436-4333 for further information on system configurations

Systems

Atmel offers several levels of systems to meet the various needs of our customers. The systems are described on the following pages, along with a brief description of their function. For further information, contact your local Atmel sales representative or the Atmel PLD Hotline at (408) 436-4333.

Table 1. PLD Software System Recommendations

User	Experience	Recommended System
New PLD User	No prior PLD experience. Wants basic support for PAL-type and V-Series devices and is willing to do manual pin assignment.	ATDS1200PC (Atmel-Abel)
PAL-type Device User	Knows Abel, Cupl, or PALASM, and is familiar with 22V10 design. Wants to move up in density, and wants automatic fitting to reduce need to learn detailed internals of each device.	ATDS1210PC (Atmel-Abel V-Series Kit)
Complex PLD User or TTL Designer	Uses complex PLDs from Altera, AMD, or Lattice. Wants state-of-the-art tools for Atmel's V-Series to take advantage of their speed and density. Wants schematic capture, simulation, and multiple design entry modes. Does not need full timing simulation	ATDS1303PC (Atmel-ViewPLD 3K Gate System. Includes one year of maintenance)
	If timing simulation and higher gate capacity is needed ...	ATDS1320PC (Atmel-ViewPLD 20K Gate System)
Viewlogic-based FPGA User	Uses FPGAs from Xilinx, Actel, or others with Viewlogic CAE tools for schematic capture and simulation. Wants to add capability for Atmel's V-Series to take advantage of their speed and density for control and timing applications, at minimum cost.	ATDS1303PCI (Atmel-ViewPLD 3K System Intermediate Upgrade)
	If timing simulation and higher gate capacity is needed ...	ATDS1320PCI (Atmel-ViewPLD 20K Gate System Intermediate Upgrade)

Note: 1. For proper ordering codes, consult the Atmel PLD Software Tools Product Description List

Table 2. PLD System Configuration Requirements

	Atmel-Abel (1)	Atmel-ViewPLD
Platform:	PC386/486	PC386/486
Operating System:	DOS 3.0 or greater	DOS 3.0 or greater
Display Adaptor:	CGA, EGA, Hercules, VGA or SVGA	EGA, Hercules, VGA, or SVGA
System Memory:	4 Mbytes total	6 Mbytes total min. 12 Mbytes total recommended
Hard Disk: Space required: Space Recommended:	5 Mbytes 40 Mbytes	26 Mbytes 80 Mbytes
I/O Ports required:	1 parallel	1 parallel 1 serial (for mouse)
Mouse:	not required	Mouse systems or compatible

PLD Software Tools Product Description List

Atmel-Abel
Ordering Code: ATDS1200PC

Basic Abel support for Atmel PAL-type and V-Series devices. Does not include any automatic device fitters, which are optional and priced separately or available in packages called Atmel-Abel Kits. Current users of Atmel-Abel-4.0 do not get an automatic upgrade to this version.

Atmel-Abel V-Series Fitters
Ordering Code: ATDS1205PC

Automatic ATV750/ATV2500, ATV750B/ATV2500B, and ATV5000/ATV5100 device fitters for Atmel-Abel. Requires Atmel-Abel version 4.2 or greater in order to run.

Atmel-Abel V-Series Kit
Ordering Code: ATDS1210PC

A package consisting of Atmel-Abel together with fitters for the ATV750/ATV2500, ATV750B/ATV2500B, and ATV5000/ATV5100. A complete development package for the PAL-type devices and the V-Series with fitters, that offers a savings to the user over the components purchased separately.

Atmel-ViewPLD 20K Gate System
Ordering Code: ATDS1320PC

A complete stand-alone ViewPLD system with all Abel functions plus Viewlogic schematic capture, simulation (unit delay and timing) as well as modeling and partitioning functions plus utilities. Includes all fitters. Price includes one year of maintenance/updates. Maintenance is optional after the first year.

Atmel-ViewPLD 3K Gate System
Ordering Code: ATDS1303PC

A complete stand-alone ViewPLD system with all Abel functions plus Viewlogic schematic capture, simulation (unit delay). Includes all fitters. Price includes one year of maintenance/updates. Maintenance is optional after the first year.

Atmel-ViewPLD Maintenance
Ordering Code: ATDM1303PC (3K)
ATDM1320PC (20K)

One year of updates and maintenance for the ViewPLD software. Maintenance for one year is included with the purchase of Atmel-ViewPLD, and it is optional after the first year. Maintenance is not required for the purchase of an Atmel-ViewPLD Intermediate Upgrade, but is available if the user wishes to purchase their updates from Atmel.

Atmel-ViewPLD Intermediate Upgrade
Ordering Code: ATDS1303PCI (3K)
ATDS1320PCI (20K)

A ViewPLD system sold as an add-on for users with an existing vendor-specific (SDA) Viewlogic seat. Examples of SDA seats are Viewlogic front-end systems sold by Xilinx and Actel for use with their place-and-route tools. Adds Atmel PLDs to the system. Includes all Atmel-Abel files and fitters plus access to the appropriate Viewlogic libraries. May not be sold to a user with a full Viewlogic seat purchased from Viewlogic. Those users must purchase ViewPLD directly from Viewlogic.

Ordering Information

Ordering Code	Description
ATDS1200PC	Atmel-Abel High-Level Design Tool for Atmel Programmable Logic Devices: PAL-type Devices, ATV750, ATV750B, ATV2500, ATV2500B, ATV5000, and ATV5100
ATDS1205PC	Atmel-Abel Custom Logic Fitter Package for V-Series CPLDs
ATDS1210PC	Atmel-Abel Kit with all V-Series Fitters
ATDS1303PC	Atmel-ViewPLD 3K Gate System with Functional Simulation for PCs (Includes One Year Maintenance)
ATDS1303PCI	Atmel-ViewPLD 3K Gate Intermediate Upgrade System with Functional Simulation
ATDM1303PC	Atmel-ViewPLD 3K Gate System Maintenance (One Year)
ATDS1320PC	Atmel-ViewPLD 20K Gate System with Timing Simulation for PCs (Includes One Year Maintenance)
ATDS1320PCI	Atmel-ViewPLD 20K Gate Intermediate Upgrade System with Timing Simulation
ATDM1320PC	Atmel-ViewPLD 20K Gate System Maintenance (One Year)

Features

- Atmel-Abel Uses the Industry-Standard Abel Hardware Description Language
- Multiple Input Methods :
  - Boolean Equations, Truth Tables and State Diagrams
  - Optional Schematic Entry Available
- Automatic Logic Reduction, Simulation, Error Checking, and Generation of Design Documentation
- Optional Device Fitters Perform Automatic Pin/Node Assignment and Logic Synthesis
- Automatically Takes Advantage of Atmel's PLD Architecture, Joining Sum Terms When Extra Product Terms are Needed
- Runs on MS-DOS™ Compatible Personal Computers
- This Inexpensive Package Includes:
  - Atmel-Abel Software which Supports Atmel PAL-type Devices, ATV750, ATV750B, ATV2500, ATV2500B, ATV5000, and ATV5100
  - Complete Abel Manual
  - Design Examples
- Upgradable to Full Version of Abel Through Data I/O

Atmel-Abel High-Level Design Tool for Atmel Programmable Logic Devices:

PAL-type

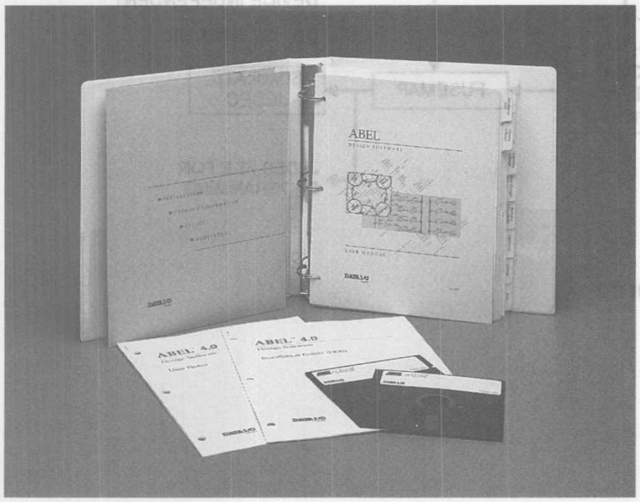
ATV750

ATV2500

ATV5000

ATV5100

3



Description

Atmel Programmable Logic Devices (PLDs) offer powerful solutions for logic design. Atmel-Abel, developed by Data I/O Corporation, is a software package specifically designed to support development with Atmel Programmable Logic Devices.

Atmel-Abel offers all of the function and features of Data I/O's standard Abel software package while supporting Atmel's PLDs including all Atmel PAL-type devices as well as the ATV750, ATV750B, ATV2500, ATV2500B, ATV5000, and ATV5100. Support for other manufacturer's PLDs is not provided.

Atmel-Abel automatically takes advantage of Atmel's innovative multiple sum term PLD architecture. When your reduced equations require more product terms than anticipated, the software automatically allocates the next available block of product terms to your equation.

Continued on next page

Description (Continued)

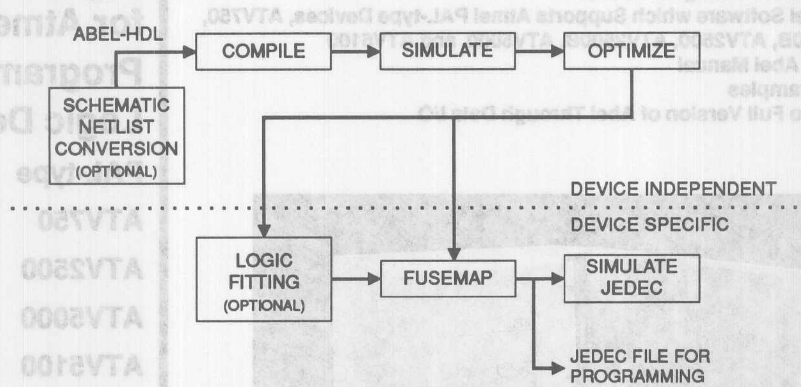
Atmel-Abel automatically reduces your logic equations to near minimal form. Depending on your requirements, you can choose among several reduction algorithms. The result is a more efficient, cost-effective design.

Behavioral simulation is an integral part of the Atmel-Abel design package. Simulation may be performed on either the design equations as entered, or after device selection, pin and node assignment, and option selection. This way you can verify that the completed design matches your design input.

Atmel-Abel uses the Open Abel format which allows the use of optional logic fitters. These logic fitters perform automatic device pin and node assignment and macrocell option selection to make maximum use of device resources, speeding up the design process.

Once your design is ready, Atmel-Abel generates standard JEDEC files which can be downloaded to your programmer with the terminal emulation software included with the package.

Design Flow Diagram



General Support Requirements

- 386/486 Computer
- 3 1/2" 1.44 MB/HD Floppy Disk Drive
- DOS 3.0 or higher
- 640K RAM, 4 MB Extended RAM

Ordering Information

Ordering Code	Description
ATDS1200PC	Atmel-Abel High-Level Design Tool for Atmel Programmable Logic Devices: PAL-type Devices, ATV750, ATV2500, ATV5000, ATV5100 and H-Series
ATDS1210PC	Atmel-Abel Kit with all V-Series Fitters

Abel™ and MS-DOS™ may be trademarks of others.

Atmel-Abel V-Series Custom Logic Fitters:

ATV750

ATV750B

ATV2500

ATV2500B

ATV5000

ATV5100

3

Features

- **Works with Atmel-Abel PLD Design System**
- **Allows Device-Independent Design**
- **Optimizes Resource Utilization**
- **Assigns Signals to Pins and Nodes**
- **Utilizes Atmel's Unique Product Term Joining/Sharing**
- **Performs Logic Synthesis**

Description

This custom logic fitter package for the Atmel-Abel PLD design system supports all Atmel V-Series CPLDs. It was written by Data I/O with Atmel's cooperation and approval. This fitter package will work only with Atmel-Abel, the special version of Abel available from Atmel which supports Atmel's low and high density PLDs.

The fitters perform pin/node signal assignment, perform macrocell option selection, take advantage of Atmel's shared product term architecture where possible, and maximize resource utilization. This allows a user to do device-independent design and still get maximum utilization for Atmel's V-Series PLDs without needing to know all internal details of the devices.

Pin/node assignments may be made by the user, assigned by the fitter, or a combination of both. Optionally, all pin/node assignments can be ignored and the fitter will attempt to fit the design into the target device automatically.

Requirements

- The V-Series fitters require Atmel-Abel version 4.2 or higher running under MS-DOS, version 3.00 or higher, with extended memory support (6 Mbyte minimum).

Ordering Information

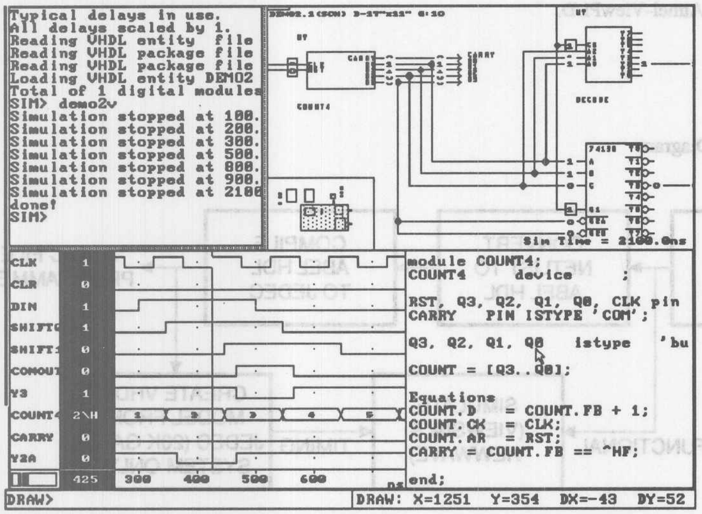
Ordering Code	Description
ATDS1205PC	Atmel-Abel Custom Logic Fitter Package for V-Series CPLDs

Reduce your time to market with the fastest, easiest to use, most complete set of tools for programmable logic design available: Atmel-ViewPLD. This fully integrated system supports Atmel's PLD devices and V-Series of complex PLDs.

Benefits

- **Allows design specification with schematic entry or the Abel hardware description language and accepts existing designs with JEDEC files and TTL logic in any arbitrary design hierarchy.**
- **Allows the retargeting of older designs into new technologies, and new functions can be easily merged and enhanced.**
- **Utilizes intelligent device fitters for automatic logic synthesis and device resource assignment.**
- **Saves design iterations by generating full VHDL timing models for simulating and debugging PLD designs in the system context.**
- **Optimizes designs requiring multiple devices with automatic user-guided partitioning and device fitting.**
- **Can combine multiple PLDs into larger CPLDs**

ViewPLD Sample Screen



Atmel-ViewPLD is a Complete Set of Tools for Programmable Logic Design

Through a partnership with Data I/O and Viewlogic Systems, Atmel has developed Atmel-ViewPLD, bringing together the best programmable device knowledge and fitting capabilities with the best engineering environment for schematic entry and verification. Atmel-ViewPLD is a complete set of tools for PLD design which will get you from concept to finished product faster than you ever thought possible. Atmel-ViewPLD contains every tool you will ever need, from schematic entry and device fitting to automatic partitioning — even waveform analysis and full timing simulation.

Atmel-ViewPLD is available either as an add-on product to users with other vendor-specific Viewlogic front-end systems (intermediate upgrade), or as a stand-alone product offering unmatched capabilities and value in PLD design.

Atmel-ViewPLD Integrated Programmable Logic Design System

ViewPLD Works From Multiple Input Formats/Sources

All of Atmel-ViewPLD's capabilities operate from a design database which is created from several sources which can be freely intermixed. The Abel language can be used to easily specify Boolean operations, truth tables, and state machines. Schematics can be created using basic logic functions or TTL macro functions, and existing PLD designs in JEDEC format can be used as well. Once the design database is created, Atmel-ViewPLD can automatically produce a schematic of the design for use in system schematics for functional simulation before any implementation is started.

Design Flow

Figure 1 shows the basic design flow for Atmel-ViewPLD. Abel source files can be processed directly using the basic technology supplied by Data I/O, or Viewlogic's ViewDraw tool may be used to enter hierarchical designs. These designs may have blocks that originate as Abel files, JEDEC files, or schematic input using either logic primitives or any of the comprehensive set of TTL macro functions included with Atmel-ViewPLD.

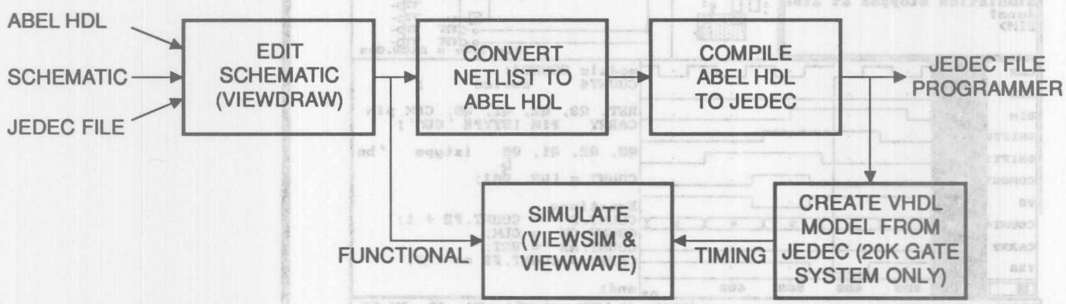
Comprehensive Device Support

Atmel-ViewPLD offers complete support for Atmel's families of PLDs, including 22V10 devices and the V-Series. The V-Series, designed for high density timing and control applications, includes the ATV750/L, ATV750B/BL, ATV2500H/L, ATV2500B/BL, ATV5000/L and ATV5100/L. For each PLD, Atmel-ViewPLD automatically produces optimized JEDEC files that are ready to be downloaded to device programmers.

Although JEDEC file generation and timing modeling is limited to the Atmel devices, JEDEC file design input into the design database is accepted for most PLD devices supported by Abel, including all PAL and GAL devices.

Atmel-ViewPLD includes device fitters for all supported devices. These fitters, developed by Data I/O with Atmel's support and cooperation, perform logic synthesis and device resource assignment automatically, allowing efficient designs to be created with minimum effort.

Figure 1. Atmel-ViewPLD Design Flow Diagram



Atmel-ViewPLD Offers Interactive Partitioning

The design database can be automatically partitioned into multiple PLDs. Using ViewDraw, Viewlogic's schematic capture tool, constraints are specified graphically, such as device pin sizes, signal grouping, and register and pin spares.

After partitioning, schematic symbols are automatically created for the devices and the partitioned schematic is automatically drawn. This can then be used to add additional constraints and iterate the partitioning process. Once partitioning is completed, each partition is then "fitted" into the best device as controlled by the designer.

VHDL Models Automatically Generate Simulation Models

Once the design has been fitted into the PLD(s), Atmel-ViewPLD generates simulation models for each PLD in VHDL for design verification. These models can be used with the automatically generated schematic interconnecting the PLD(s) to perform full timing simulation at the system level using ViewSim and ViewWave (Viewlogic's full function simulator and graphic waveform analyzer). ViewSim and ViewWave are included as a part of Atmel-ViewPLD.

Functional VHDL models can also be produced for use in migrating a design to ASIC technology using the optional View-Synthesis and ViewRetargeter tools.

Ordering Information

Ordering Code	Description
ATDS1303PC	Atmel-ViewPLD 3K Gate System with Functional Simulation for PCs (Includes One Year Maintenance)
ATDS1303PCI	Atmel-ViewPLD 3K Gate Intermediate Upgrade System with Functional Simulation
ATDM1303PC	Atmel-ViewPLD 3K Gate System Maintenance (One Year)
ATDS1320PC	Atmel-ViewPLD 20K Gate System with Timing Simulation for PCs (Includes One Year Maintenance)
ATDS1320PCI	Atmel-ViewPLD 20K Gate Intermediate Upgrade System with Timing Simulation
ATDM1320PC	Atmel-ViewPLD 20K Gate System Maintenance (One Year)

Software Support Information(1)

Software Versions	Software						
	Data I/O Abel™	Logical Devices Cupl™	ISDATA LOG/iC	Atmel Abel™	Atmel ViewPLD	View Logic ViewPLD	Minc[4] PL Designer
AT18V8Z	3.2	2.1	3.0	4.40	4.40	1.2	1.5
AT22LV1	1.3, 2.0, 3.0, 4.0	2.0	2.3	4.4	4.4	1.2	1.5
ATF22V10B	1.3, 2.0, 3.0, 4.0	2.0	2.3	4.4	4.4	1.2	1.5
AT22V10B	1.3, 2.0, 3.0, 4.0	2.0	2.3	4.4	4.4	1.2	1.5
AT22V10	1.3, 2.0, 3.0, 4.0	2.0	2.3	4.4	4.4	1.2	1.5
ATV750	3.0	2.15b	3.0	4.2	4.2	1.2	3.0[1]
ATV2500	3.2	3.2A	3.4	4.2	4.2	1.2	3.0[1]
ATV5000	4.02	4.3		4.2	4.2	1.2	(3)
ATV5100	4.3			4.3	4.3	4.3	(3)
ATF16V8B	2.0	2.0	2.3	4.40	4.40	1.2	1.5
ATF20V8B	2.0	2.0	2.3	4.40	4.40	1.2	1.5
ATV750B	5.1[2]	4.4c	(3)	4.40	4.40	(3)	3.2e
ATV2500B	5.1[2]	4.4c	(3)	4.41	4.41	(3)	3.2e

Notes:
1. Supported in Minc PLDesigner - XL only.
2. Scheduled release.
3. Under development. Please call PLD software manufacturer for updated information.
4. Minc tool is an OEM for CAE tools from Cadence, Mentor and Racal Redac. Synopsys users get support from Minc's library.

CAE Software Companies

Data I/O Corporation (Abel™)
10525 Willows Rd. N.E.
P.O. Box 97046
Redmond, WA 98073-9746
(206) 881-6444
(800) 247-5700

ISDATA Gmbh (LOG/iC)
Daimlerstr 51
W-7500 Karlsruhe 21
Germany
US: (510) 531-8553

Logical Devices (Cupl™)
692 S. Military Trail
Deerfield Beach, Florida 33442
(305) 428-6868

Minc, Inc.
6755 Earl Drive
Colorado Springs, CO 80918
(719) 590-1155

Viewlogic Systems, Inc.
293 Boston Road West
Marlboro, MA 01752
(508) 480-0881

CMOS PLD Programming Hardware and Software Support

3

New Programmer Support Information

The following is a partial listing of the ever-growing group of programmers that are evaluated and approved by Atmel.

A list of specific product and programmer firmware/software revisions that support each product is published periodically by Atmel.

Information regarding specific product support information may be obtainable through your local Atmel representative or directly from the programmer vendors.

Company	Model						
Advin Systems	Sailor-PAL	Pilot-U40	Pilot-U84				
BP Microsystems	PLD1100	BP-1200					
Bytek	135H/ UNICEL300						
CEIBO	MP-51						
Data I/O	Unisite 48	29B	DIP Module 303A011A	PLCC Module	60A	3900	2900
Hi-Lo System	All-03A	S.W. 8.3					
Inlab/ProLogic	28A						
Logical Devices	AllPro40	AllPro88					
Micropross	ROM 3000U						
Owen	AP-II						
PistoHI	PET100						
SMS	Sprint Plus	Sprint Expert					
Stag	ZL30A						
Strebor	PLP-S1						
System General	SGUP-85	TURPRO-1					

Programming Hardware Companies

Advin Systems, Inc.
1050-L East Duane Ave.
Sunnyvale, CA 94086
(408) 243-7000

BP Microsystems
1000 N. Post Oak Road
Suite 225
Houston, TX 77055
(713) 688-4600

BYTEK
543 N. W. 77 Street
Boca Raton, FL 33487
(407) 994-3520

CEIBO
Merkazim Building
1 Maskit Street
P.O. Box 2106
Herzelia 46120
Israel
(972) 52-555387
US: (617) 863-9927

Data I/O Corporation
10525 Willows Rd. N.E.
P.O. Box 97046
Redmond, WA 98073-9746
(206) 881-6444
(800) 247-5700

Elan Digital Systems Ltd.
Elan House
Little Park Farm Road
Segensworth West
Fareham, Hants
U.K. PO15-5SJ
(0489) 579799

HI-LO System
44388 Grimmer Blvd.
Fremont, CA 94538
(510) 623-8859

Inlab / ProLogic
557-0 Burbank Street
Broomfield, CO 80020
(800) 237-6759

Logical Devices
692 S. Military Trail
Deerfield Beach, FL 33442
(305) 974-0967

Micropross
5 Rue Denis Papin
59650 Villeneuve D'Ascq
France
(011) 33204-79040

Owen
Ringstr. 11
Postfach 1104
D-6798 Kusel
Germany
(011) 49-6381-5085

SMS SPRINT International
PO Box 3159
Redmond, WA 98052-3159
US: (206) 883-8447

Strebor
1008 N. Nob Hill Dr.
America Fork, UT 84003
(801) 756-3605

Stag Microsystems
1600 Wyatt Dr.
Santa Clara, CA 95054
(408) 988-1118

System General
510 S. Park Victoria Drive
Milipitas, CA 95035
(408) 263-6667
Taiwan: (886) 2-9173005

FPGA Integrated Development System Overview

Features

- Graphical User Interface
- Unified Design Database
- Macro Library of Over 200 Hard/Soft Functions
- Viewlogic™ Viewdraw™ Schematic Capture
- Automatic Place and Route
- Interactive Layout Editing
- Viewlogic Viewsim™ Simulation
- Post-Layout Timing Analysis
- Critical Paths Shown in Schematic and Layout
- Delay Back-Annotation to Simulation

Description

Atmel's Integrated Development System lets designers create fast, predictable designs with AT6000 Series FPGAs.

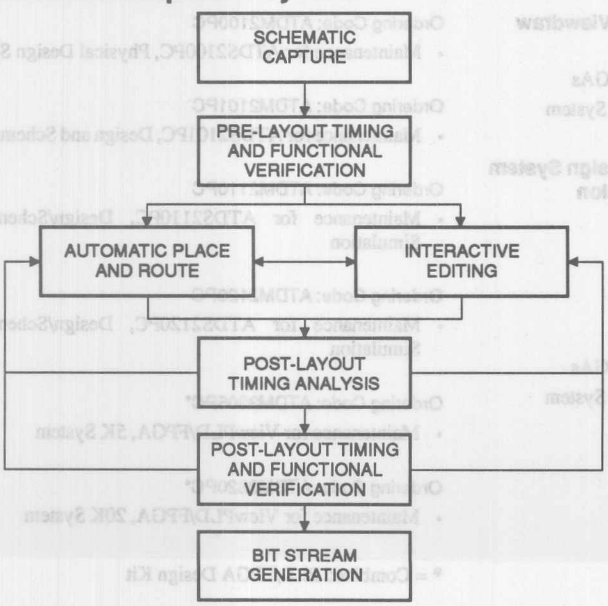
Available for use on an 386/486-based computer, the Integrated Development System combines industry-standard software for design entry and simulation with Atmel's proprietary software for automatic and interactive placement and routing, timing analysis and bit stream generation.

The Integrated Development System design flow is shown below. Pre-layout modules verify design logic, place and route modules implement the design in silicon. Post-layout modules reflect the design as it actually appears in silicon.

A Design Manager provides push-button access to each step in the flow. The Design Managers simple user interface streamlines the design flow as it creates a seamless design environment. Design data is stored in a unified database that eliminates the need for data re-entry and translation.

The Integrated Development System is available in a low-cost package and includes timing analysis and a prototype kit. Timing and functional simulation is optional.

Integrated Development System



3

AT6000 FPGA System Summary

The following is a summary of Integrated Development System software, hardware and annual maintenance agreements. For availability and pricing, contact your local Atmel sales representative or distributor.

FPGA Design Tools

ATDS2100PC FPGA Physical Design System

Ordering Code: ATDS2100PC

- Atmel Design Manager
- AT6000 Macro Library for Viewlogic
- Netlist Generator
- Automatic Place and Route
- Interactive Editor
- Static Timing Analyzer
  - Pre-Layout Timing Estimator
  - Post-Layout Timing Analyzer
  - Timing to Simulation Back Annotator
  - Timing to Layout Back Annotator
  - Incremental Design Change
  - User Macro Library Management
- Design Rule Checker
- Layout vs. Schematic Checker
- Plotting Utilities
- System Utilities
- Bit Stream Generator
- Prototype Kit (ATDH2080 - Not included in University System)

ATDS2101PC FPGA Design System with Viewdraw

Ordering Code: ATDS2101

- Viewdraw Schematic Capture for AT6000 FPGAs
- Complete ATDS2100 FPGA Physical Design System

ATDS2110PC and ATDS2120PC FPGA Design System with Viewdraw and 10K/20K Gate Simulation

Ordering Codes: ATDS2110, ATDS2120

- Viewsim Functional Simulator
- Simulation Netlist Generator
- Viewsim Timing Simulator
- Simulation Utilities
- Viewdraw Schematic Capture for AT6000 FPGAs
- Complete ATDS2100 FPGA Physical Design System

Download Board Kits

Atmel offers download boards for use with the Prototype Kit (included in the ATDS2100 Design Package) and the designer's target system. Each kit includes a cable for downloading configuration data to a device and a socket assembly to plug the FPGA into a printed circuit board.

ATDH2084 84-Pin Download Kit

Ordering Code: ATDH2084

- 84-Pin PLCC Package Adapter Module
- 25-Pin Connector Cable
- Cable Adapter with Schmitt Trigger Module

ATDH2132 132-Pin Download Kit

Ordering Code: ATDH2132

- 132-Pin PQFP Package Adapter Module
- 25-Pin Connector Cable
- Cable Adapter with Schmitt Trigger Module

Annual Maintenance Agreements

Annual Maintenance Agreements are available for each package and option in the Integrated Development System. The first year of maintenance is included in the purchase price; renewal is optional. Maintenance Agreements give users direct access to Atmel's experienced technical support staff and cover software upgrades that keep engineers on the leading edge of Atmel's design tools.

Ordering Code: ATDM2100PC

- Maintenance for ATDS2100PC, Physical Design System

Ordering Code: ATDM2101PC

- Maintenance for ATDS2101PC, Design and Schematic Entry

Ordering Code: ATDM2110PC

- Maintenance for ATDS2110PC, Design/Schematic/10K Simulation

Ordering Code: ATDM2120PC

- Maintenance for ATDS2120PC, Design/Schematic/20K Simulation

Ordering Code: ATDM3305PC*

- Maintenance for ViewPLD/FPGA, 5K System

Ordering Code: ATDM3320PC*

- Maintenance for ViewPLD/FPGA, 20K System

* = Combined PLD/FPGA Design Kit

Hardware Requirements

- 386/486-Based Computer
- MS-DOS 3.0 or Higher
- 640 Kilobytes Base-System RAM
- 8 Megabytes Extended RAM
- 40 Megabyte (Minimum) Hard-Disk Space
- VGA Adaptor and Display
- One Serial Port
- One Parallel Port
- One Mouse Systems or Compatible Three-Button Optical Mouse

Ordering Information

Ordering Code	Description
ATDS2100PC	FPGA Physical Design System
ATDS2101PC	FPGA Design System with Viewdraw
ATDS2110PC	FPGA Design System with Viewdraw and 10K Gate Simulation
ATDS2110PCI	FPGA Design System, Viewlogic 10K Upgrade
ATDS2120PC	FPGA Design System with Viewdraw and 20K Gate Simulation
ATDS2120PCI	FPGA Design System, Viewlogic 20K Upgrade
ATDH2080	FPGA Prototype Kit (included with ATDS2100PC)
ATDH2084	84-Pin Download Kit
ATDH2132	132-Pin Download Kit
ATDM2100PC	Maintenance for Atmel FPGA, Physical Design System
ATDM2101PC	Maintenance for Atmel FPGA, Design and Schematic Entry
ATDM2110PC	Maintenance for Atmel FPGA, Design/Schematic/10K Simulation
ATDM2120PC	Maintenance for Atmel FPGA, Design/Schematic/20K Simulation
ATDM3305PC	Maintenance for Atmel ViewPLD/FPGA, 5K System
ATDM3320PC	Maintenance for Atmel ViewPLD/FPGA, 20K System

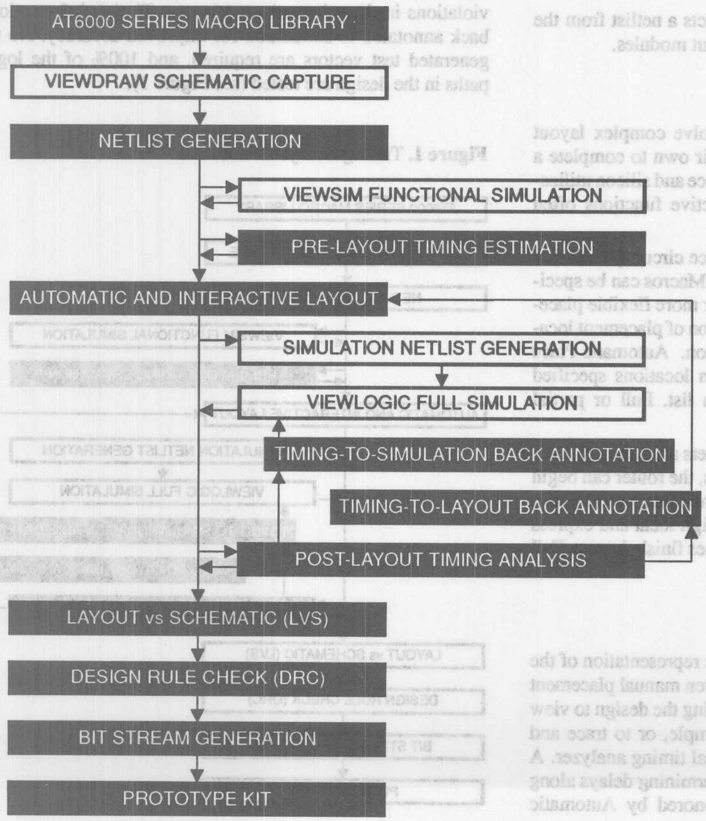
FPGA Physical Design System

3

Features

- A Low-Cost Package for Completing an AT6000 Series Design
- Atmel Design Manager
- AT6000 Macro Library for Viewlogic
- Netlist Generator
- Automatic Place and Route
- Interactive Editor
- Static Timing Analyzer (Circuit Verification Tools)
  - Pre-Layout Timing Estimator
  - Post-Layout Timing Analyzer
  - Timing to Simulation Back Annotator
  - Timing to Layout Back Annotator
  - Incremental Design Change
  - User Macro Library management
- Design Rule Checker
- Layout versus Schematic Checker
- Plotting Utilities
- System Utilities
- Bit Stream Generator
- Prototype Kit (ATDH2080 - Not Included in University System)

ATDS2100PC Design Flow



Description

AT6000 Series Design Manager

Operation of the Integrated Development System is controlled from a single graphical interface, the Design Manager. Similar to a control panel in format, the Design Manager displays a series of buttons, with each button corresponding to a different phase in the design process. Any module, be it schematic entry, place and route or bit stream generation, is invoked by the press of a button. The Design Manager lets the user move freely between modules and make changes at any stage in the development cycle. Data is linked to a common database which eliminates the need for data re-entry.

AT6000 Series Macro Library

The AT6000 Series Macro Library contains over 200 fully-characterized functions ranging in size from basic gates to complex functions and I/O. Alternate layouts, optimized for performance or silicon utilization, are provided for most macros. An index appears at the back of this section (see Macro Library Listing).

Netlist Generator

The Netlist Generator automatically extracts a netlist from the Viewdraw schematic for use with the layout modules.

Automatic Place and Route

The Automatic Place and Route tools solve complex layout problems quickly and can be used on their own to complete a layout. But, for optimum circuit performance and silicon utilization, a mixture of automatic and interactive functions often works best.

Automatic Place follows the netlist to place circuitry such that routing is as easy to complete as possible. Macros can be specified hard for predictable timing, or soft for more flexible placement. The program supports back annotation of placement locations to the schematic for easy identification. Automatic Place can select I/O pin placements or use pin locations specified manually in the top-level Viewdraw pin list. Full or partial placement is also supported.

The Automatic Route program can route nets according to user-specified priorities. Over a series of passes, the router can begin with critical nets and work its way up to the least critical paths, or vice versa. Global routing routines assign local and express buses to carry signals; detail routing routines finish the rest. Full or partial routing is also supported.

Interactive Editor

The Interactive Editor provides a graphic representation of the AT6000 series architecture for netlist-driven manual placement and routing. It can be used at any time during the design to view a particular cell instance or net, for example, or to trace and modify critical paths found by the optional timing analyzer. A timing calculator is also available for determining delays along any user-selected path. Changes are honored by Automatic Place and Route, so users can optimize portions of a design and leave the rest to the automatic tools.

The Interactive Editor preserves labels, macro names and pin definitions from the schematic, and enforces AT6000 series design rules. Keystroke equivalents are available for most commands, including move, copy, cut, paste, shift, rotate and flip. A cumulative undo function keeps track of design changes.

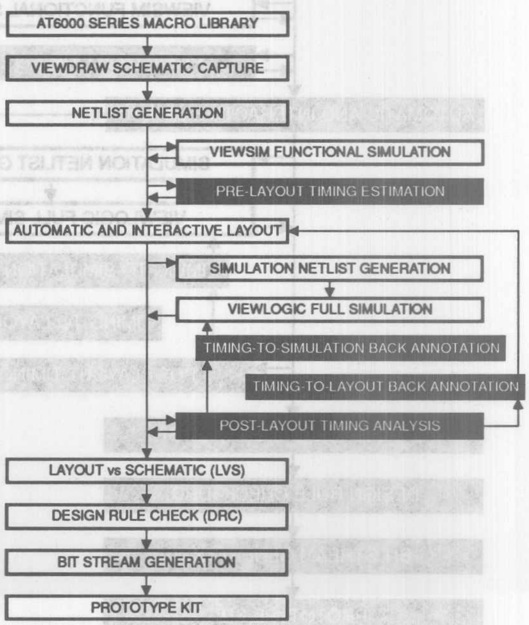
Circuit Verification Tools

The ATDS2100 Physical Design System includes timing analysis tools:

- Pre-Layout Timing Estimator
- Post-Layout Timing Analyzer
- Timing to Simulation Back Annotator
- Timing to Layout Back Annotator
- Incremental Design Change
- User Macro Library Management

The Timing Analysis Tools include a comprehensive set of analysis functions that use actual wire lengths and loading delays to identify and highlight critical paths, setup and hold time violations in the schematic and layout. Timing information is back annotated to simulation for improved accuracy. No user generated test vectors are required, and 100% of the logical paths in the design are tested (see Figure 1).

Figure 1. Timing Analysis Functions



Pre-Layout Timing Estimator

The Timing Estimator uses worst-case delays to flag setup and hold time violations in a schematic representing up to 10,000 gates of logic. Critical paths are also identified and highlighted directly in the schematic. Shortest and longest path commands assist in design analysis.

Post-Layout Timing Analyzer

The Timing Analyzer checks to see if the layout process has affected any timing dependencies. It uses actual wire lengths and loading delays to calculate post-layout timing for back annotation to the schematic, layout and simulation modules. Setup and hold time violations are identified and critical paths are highlighted directly in the layout.

Timing to Simulation Back Annotator

Post-layout timing information, including pin-to-pin delays with actual wire delays, is back annotated to Viewsim. Subsequent simulation runs reflect actual device timing.

Timing to Layout Back Annotator

Post-layout timing paths, including critical paths and setup and hold time violations, is back annotated to the Interactive Editor and Viewdraw. Shortest and longest delay paths can be traced in the layout or schematic, making it possible to improve design timing from either module.

Incremental Design Change

This function allows modifications to the schematic to be reflected in the layout. It maintains all previous placement and routing unaffected by the change which preserves existing circuit timing.

User Macro Library Management

The Macro Library allows the user to add, delete, copy and modify components for a user macro library.

Design Rule Checker

The Design Rule Checker searches for electrical and design-rule violations, reporting errors in a text file for easy debug.

Layout vs. Schematic Checker

The Layout vs. Schematic Checker compares the layout to the schematic and reports any discrepancies.

Plotting Utilities

Atmel's Plotting Utilities produce PostScript and HP-GL/2 plots of design layouts for use in device specifications and revision control.

System Utilities

System Utilities structure the work environment to suit personal work habits. Options include selecting an active design, specifying design backup procedures, restoring archived files, and revising user and design settings.

Bit Stream Generator

The Bit Stream Generator produces the configuration file used to program a AT6000 Series device. Six configuration modes are available. Configuration windows are generated according to user specification. Full and partial configuration algorithms can be generated. Simultaneous configuration of many devices is also possible.

Prototype Kit (ATDH2080)

The Prototype Kit includes a wire-wrap printed circuit board assembly with sockets that hold 84- and 132-pin packages. A 25-pin connector cable and cable adaptor are provided for attaching the board to a personal computer. The board can be used for quick, convenient device configuration, or populated with active components to test the FPGA in a systems environment.

Note About Data Bases

The Integrated Development System uses a common data base structure that eliminates the need for data re-entry and manual translation.

The netlist generation, macro library and user-defined component data contained in the design schematic is stored in a logic design data base. The simulation and timing analysis programs use this data to perform pre-layout functions.

When the layout programs are invoked, the data in the logic design database is automatically transferred to a layout design data base for use with automatic and interactive placement and routing routines. The layout design data base is also used for post-layout simulation, timing analysis and error checking.

This common data base approach streamlines the design flow while it ensures that design data remains consistent.

Rather than spend time creating simple functions from scratch, designers can select ready-made circuits from the AT6000 Series Macro Library, (included in the ATDS2100 Development System) for use in their schematic or layout. Listed below are the more than 200 functions contained in the Macro Library.

Macros range in size from basic gates to complex functions and I/O. Each macro is fully characterized, with schematic symbol, simulation representations, and timing analysis data. Alternate layouts, optimized for performance or silicon utilization, are provided for many macros.

Macros can be specified as hard or soft, so the user has more control over automatic placement and routing.

A reference manual is provided with the Macro Library. It includes primitive count, timing data, truth tables, schematic, symbol and layout.

Logical Function	Description	Area (XxY)
Gates - NAND		
ND2	2-Input NAND	1x1
ND3	3-Input NAND	1x1
ND4	4-Input NAND	2x1
ND5	5-Input NAND	2x1
ND6	6-Input NAND	3x1
ND8	8-Input NAND	4x1
NDND	Two 2-Input NANDs	1x1
Gates - AND		
AN2	2-Input AND	1x1
AN3	3-Input AND	1x1
AN4	4-Input AND	2x1
AN5	5-Input AND	2x1
AN6	6-Input AND	3x1
AN8	8-Input AND	4x1
AN2L	2-Input AND (AB)	1x1
AN2INV	2-Input AND with Inverter (A•L)L'	1x1
ANXO	2-Input AND Feeding an XOR	1x1
Gates - NOR		
NR2	2-Input NOR	2x1
NR3	3-Input NOR	3x1
NR4	4-Input NOR	4x1
NR5	5-Input NOR	5x1
NR6	6-Input NOR	6x1
NR8	8-Input NOR	8x1
NR2L	2-Input NOR with Inverted Input (AB')	1x1
Gates - OR		
OR2	2-Input OR	3x1
OR3	3-Input OR	4x1
OR4	4-Input OR	5x1
OR5	5-Input OR	6x1
OR6	6-Input OR	7x1
OR8	8-Input OR	9x1
ORL	2-Input OR (ANB) XOR AN	1x1
OR2L	2-Input OR (A+B')	2x1
ORT	2-iNPUT or = (A•L') XOR B	1x1

Logical Function	Description	Area (XxY)
Gates - XNOR		
XO2	2-Input XOR	1x1
XO3	3-Input XOR	2x1
XO4	4-Input XOR	2x2
XOND	2-Input XOR/NAND	1x1
XOND3	2-Input XOR with 3-Input NAND	1x1
XN2	2-Input XNOR	2x1
XN3	3-Input XNOR	3x1
XN4	4-Input XNOR	3x2
Inverters and Buffers		
INV	Inverter	1x1
INVINV	Two Inverters	1x1
INVAN2	Inverter and 2-Input AND L', (A•L)	1x1
AO22	AND-OR Inverter 2-2 Inputs	2x2
AOI22	AND-OR Inverter 2-2 Inputs	2x2
OAI22	OR-AND Inverter 2-2 Inputs	2x4
BUF	Buffer	1x1
Constants		
ONE	Logic One	1x1
ZERO	Logic Zero	1x1
ONEONE	Two Logic Ones	1x1
ZEROZERO	Two Logic Zeros	1x1
ZEROONE	Logic One and Logic Zero	1x1
Multiplexers		
MUX21	2-to-1 Multiplexer	1x1
MUX31	3-to-1 Multiplexer	4x3
MUX41	4-to-1 Multiplexer	5x4
MUX81	8-to-1 Multiplexer	7x9
MUX	2-to-1 Multiplexer (Fast)	1x1
MUX3	3-to-1 Multiplexer (Fast)	3x1
MUX4	4-to-1 Multiplexer (Fast)	2x2
MUX8	8-to-1 Multiplexer (Fast)	3x4
SELBUFS	Select Buffer	1x1
SELBUFX	Select Buffer	1x1

FPGA Macro Library (Continued)

Logical Function	Description	Area (XxY)
Decoders		
DC24	2-to-4 Decoder Active High/Low	2x5
DC24L	2-to-4 Decoder Active Low	1x6
DC24H	2-to-4 Decoder Active High	2x2
DC38L	3-to-8 Decoder Active Low	4x8
DC38H	3-to-8 Decoder Active High	3x8
EN42	4-to-2 Encoder	2x3
EN83	8-to-3 Encoder	2x3
BCD7SEG	7-Segment BCD Display (0 to A)	4x21
DIS7SEG	7-Segment BCD Display (0 to F)	5x26
Latches		
LD	D Latch Transparent High	3x2
LDL	D Latch Transparent Low	4x2
LDR	D Latch Transparent High, Reset Low	4x2
LDS	D Latch Transparent High, Set Low	4x2
LSRND	SR Latch with NAND	2x1
LSREN	SR Latch with NAND with Enable	2x2
LSRNR	SR Latch with NOR	2x2
Flip-Flops		
FD	D Flip-Flop Synchronous	1x1
FDE	D Flip-Flop Synchronous with Enable	2x1
FDEL	D Flip-Flop Synch with Enable Low	2x1
FDS	D Flip-Flop Synchronous Set Low	1x1
FDR	D Flip-Flop Synchronous Reset Low	2x1
FDSR	D Flip-Flop Synch Set/Reset Low	2x1
FDSA	D Flip-Flop Asynchronous Set Low	2x1
FDN	D Flip-Flop Synchronous with QN Out	1x1
FDXO	D Flip-Flop with XOR	1x1
FDHA	D Flip-Flop Half-Adder Sum	1x1
FDOR	D Flip-Flop 2-Input OR	1x1
FDND	D Flip-Flop 2-Input NAND	1x1
FDORL	D Flip-Flop (A+L')	1x1
FDMUX	MUX Feeding D Flip-Flop	1x1
FDXOAN3	(AL) XOR B; B = (A•L) AND B	1x1
FJK	JK Flip-Flop Synchronous	2x4
FJKQ	JK Flip-Flop Synchronous with Q Low	3x4
FJKS	JK Flip-Flop Synchronous Set Low	3x4
FJKR	JK Flip-Flop Synchronous Reset Low	3x4
FJKSR	JK Flip-Flop Synch Set/Reset Low	3x4
FJKSA	JK Flip-Flop Asynchronous Set Low	3x4
FJKRA	JK Flip-Flop Asynchronous Reset Low	2x4
FT	T Flip-Flop Synchronous	2x1
FTS	T Flip-Flop Synchronous Set Low	2x1

Logical Function	Description	Area (XxY)
Flip-Flops (Continued)		
FTR	T Flip-Flop Synchronous Reset Low	2x1
FTSR	T Flip-Flop Synchronous Set/Reset Low	2x2
FTRA	T Flip-Flop Asynchronous Reset Low	2x1
CLKEDGE	Clock Edge Detect Q=1 on Rise	1x1
Counters		
CR0	0-Bit Ripple-Carry Counter	2x1
CR1	1-Bit Ripple-Carry Counter	2X1
CR4	4-Bit Ripple-Carry Counter	2x4
CRST	Bit Stage Ripple-Carry Counter	2x1
CRP4	4-Bit Ripple-Carry with Parallel Load	2x4
CRPST	Bit Stage Ripple-Carry with Parallel Load	2x1
CD4	4-Bit Decade Counter	4x5
CDP4	4-Bit Decade Counter with Parallel Load	3x8
CJ2	2-Bit Johnson Counter	1x2
CJ3	3-Bit Johnson Counter	2x2
CJ4	4-Bit Johnson Counter	2x2
CJ5	5-Bit Johnson Counter	2x3
CJ6	6-Bit Johnson Counter	2x3
CJ8	8-Bit Johnson Counter	2x4
CJ12	12-Bit Johnson Counter	2x6
CJE2	2-Bit Johnson Counter with Enable	3x2
CJE3	3-Bit Johnson Counter with Enable	4x2
CJE4	4-Bit Johnson Counter with Enable	4x3
CJE5	5-Bit Johnson Counter with Enable	4x3
CJE6	6-Bit Johnson Counter with Enable	4x4
CJE8	8-Bit Johnson Counter with Enable	4x5
CJE12	12-Bit Johnson Counter with Enable	4x7
Adders		
FA1	1-Bit Full Adder	2x2
HA1	1-Bit Half Adder	2x1
HAL1	1-Bit Half Adder Carry Low	1x1
EC2	2-Bit Equality Comparator	3x2
EC3	3-Bit Equality Comparator	3x3
EC4	4-Bit Equality Comparator	3x4
EC8	8-Bit Equality Comparator	7x4
MC2	2-Bit Magnitude Comparator	3x7
Shift Registers		
SR2	2-Bit Shift Register	1x2
SR3	3-Bit Shift Register	1x3
SR4	4-Bit Shift Register	1x4
SR8	8-Bit Shift Register	1x8

3

FPGA Macro Library (Continued)

Logical Function	Description	Area (XxY)
Shift Registers (Continued)		
SRP2	2-Bit Shift Register with Parallel Load	1x2
SRP3	3-Bit Shift Register with Parallel Load	1x3
SRP4	4-Bit Shift Register with Parallel Load	1x4
SRP8	8-Bit Shift Register with Parallel Load	1x8
SRPST	Bit Stage Shift Register with Parallel Load	1x1
Converters		
PSC1	1-Bit Parallel-to-Serial Converter	1x1
PSC8	8-Bit Parallel-to-Serial Converter	1x8
SPC1	1-Bit Serial-to-Parallel Converter	2x2
SPC8	8-Bit Serial-to-Parallel Converter	4x8
Arithmetic Functions		
PCOE4	4-Bit Parity Checker Odd/Even	3x2
PCOE8	8-Bit Parity Checker Odd/Even	4x3
PCOE9	9-Bit Parity Checker Odd/Even	4x4
PCOE13	13-Bit Parity Checker Odd/Even	5x4
PCOE16	16-Bit Parity Checker Odd/Even	6x5
PCOE17	17-Bit Parity Checker Odd/Even	6x5
Register Files		
R2	2-Bit Register File	1x2
R3	3-Bit Register File	1x3
R4	4-Bit Register File	1x4
R8	8-Bit Register File	1x8
R4X2	4 Word x 2 Bit Register File	1x8
R4X4	Word x 4 Bit Register File	1x17
Tri-state Functions		
BUFZ	Tri-state Buffer	1x1
INVZ	Tri-state Inverter	2x1
FDZ	D Flip-Flop with Tri-state Out	1x1
FDZQ	D Flip-Flop with Tri-state Out and Q Out	1x1
R2Z	2 Bit Register File with Tri-state Out	1x2
R3Z	3 Bit Register File with Tri-state Out	1x3
R4Z	4 Bit Register File with Tri-state Out	1x4
R8Z	8 Bit Register File with Tri-state Out	1x8
HZ	Bus Driver High or Z	1x1
LZ	Bus Driver Low or Z	1x1
CLKEDGEZ	Clock Edge Detect Q=1 on Rise (Z)	1x1

Logical Function	Description	Area (XxY)
Input/Outputs		
Inputs		
ICMS	Input Buffer CMOS Level	1x1
ITTL	Input Buffer TTL Level	1x1
ICMSP	Input Buffer CMOS Level with Pull-Up	1x1
ITTLP	Input Buffer TTL Level with Pull-Up	1x1
Outputs		
OD	Output Buffer Drive	1x1
ODP	Output Buffer Drive with Pull-Up	1x1
ODF	Output Buffer Drive Fast	1x1
ODPF	Output Buffer Drive with Fast Pull-Up	1x1
OOC	Output Buffer Open Collector	1x1
OP	Output Buffer Open Collector with Pull-Up	1x1
OOCF	Output Buffer Open Collector Fast	1x1
OPF	Output Buffer Open Collector with Fast Pull-Up	1x1
Tri-state Outputs		
ODEN	Tri-state Output Drive	1x1
ODPEN	Tri-state Output with Pull-up	1x1
ODFEN	Tri-state Output Drive Fast	1x1
ODPFEN	Tri-state Output with Fast Pull-Up	1x1
OOCEN	Tri-state Output Open Collector	1x1
OPEN	Tri-state Output Open Collector with Pull-Up	1x1
OOCFEN	Tri-state Output Open Collector Fast	1x1
OPFEN	Tri-state Otpt Open Collector with Fast Pull-Up	1x1
Bi-directional I/O		
BTTLD	Bi-directional I/O TTL Drive	1x1
BCMSD	Bi-directional I/O CMOS Drive	1x1
BTTLDF	Bi-directional I/O TTL Drive Fast	1x1
BCMSDF	Bi-directional I/O CMOS Drive Fast	1x1
BTTLDP	Bi-directional I/O TTL Drive with Pull-Up	1x1
BCMSDP	Bi-directional I/O CMOS Drive with Pull-Up	1x1
BTTLDPF	Bi-directional I/O TTL Drive with Fast Pull-Up	1x1
BCMSDPF	Bi-directional I/O CMOS Drive with Fast Pull-Up	1x1

FPGA Macro Library (Continued)

Logical Function	Description	Area (XxY)
Bi-directional I/O with Open Collector		
BTTLOC	Bi-directional I/O TTL Open Collector	1x1
BCMSOC	Bi-directional I/O CMOS Open Collector	1x1
BTTLP	Bi-directional I/O TTL Open Collector with Pull-Up	1x1
BCMSP	Bi-directional I/O CMOS Open Collector with Pull-Up	1x1
BTTLOCF	Bi-directional I/O TTL Open Collector Fast	1x1
BCMSOCF	Bi-directional I/O CMOS Open Collector Fast	1x1
BTTLPF	Bi-directional I/O TTL Open Collector with Fast Pull-Up	1x1
BCMSPF	Bi-directional I/O CMOS Open Collector Fast Pull-Up	1x1

Logical Function	Description	Area (XxY)
Bi-directional I/O with Open Collector and Enable		
BTOCEN	Bi-directional I/O TTL Open Collector with Enable	1x1
BCOCEN	Bi-directional I/O CMOS Open Collector with Enable	1x1
BTPEN	Bi-directional I/O TTL Open Collector Pull-Up Enable	1x1
BCPEN	Bi-directional CMOS Open Collector Pull-Up Enable	1x1
BTOCFEN	Bi-directional I/O TTL Open Collector Fast Enable	1x1
BCOCFEN	Bi-directional I/O CMOS Open Collector Fast Enable	1x1
BCPFEN	Bi-directional CMOS Open Collector Fast Pull-Up Enable	1x1
BTPFEN	Bi-directional TTL Open Collector Fast Pull-Up Enable	1x1

3

Ordering Information

Ordering Code	Description
ATDS2100PC	FPGA Physical Design System

Features

- **Viewlogic Schematic Capture for AT6000 Series FPGAs**
- **Complete ATDS2100PC FPGA Physical Design System**

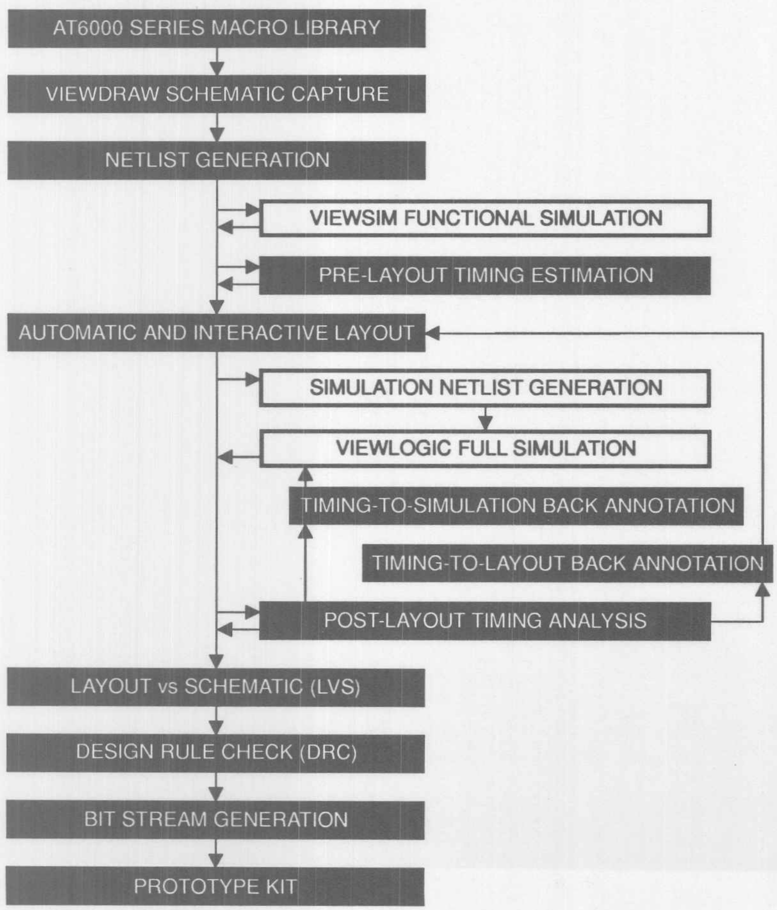
Description

The ATDS2101PC is a low-cost package for completing an AT6000 series design for users who want to use Viewdraw Schematic Capture for design entry. The ATDS2101PC contains all the features of the ATDS2100PC with the addition of Viewdraw Schematic Capture.

Viewdraw Schematic Capture

Viewdraw uses a graphic format that accepts AT6000 Series macros and supports unlimited levels of design hierarchy. Viewdraw can cut and paste between schematics, code keystroke equivalents for menu commands and plot schematics. It offers an on-line symbol editor, supports intelligent rubber-banding of nets and displays back-annotated simulation results.

ATDS2101PC Design Flow with Viewdraw



FPGA Design System with Viewdraw

Ordering Information

Ordering Code	Description
ATDS2101PC	FPGA Physical Design System with Viewdraw

FPGA Design System with Viewdraw and Viewsim

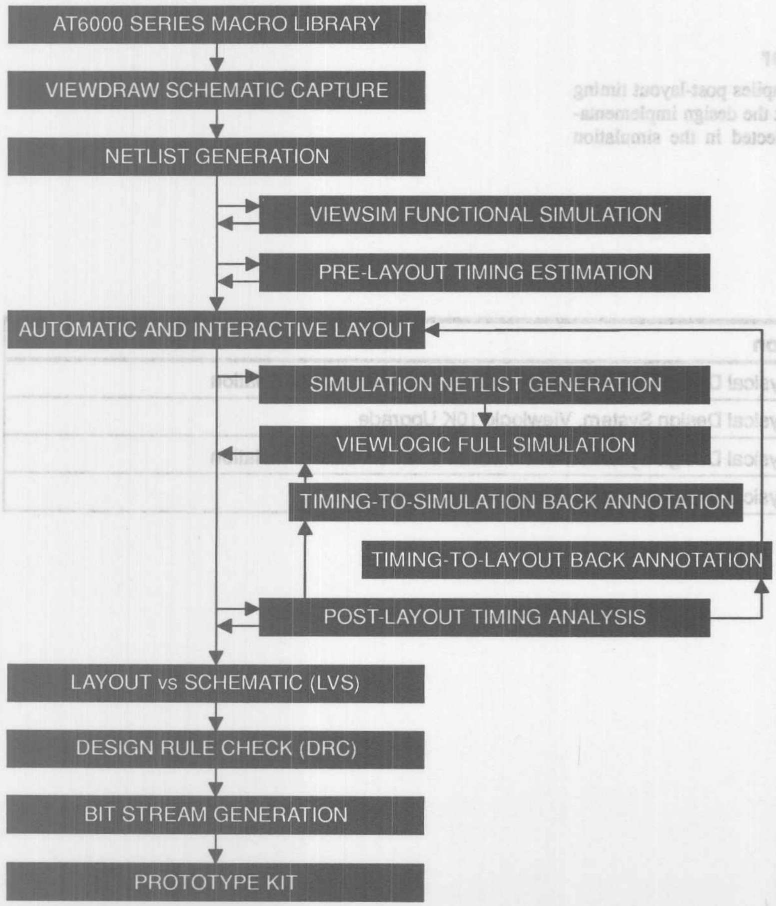
Features

- Viewsim Functional Simulator
- Simulation Netlist Generator
- Viewsim Full Simulation
- Simulation Utilities
- Viewdraw Schematic Capture for AT6000 FPGAs
- Complete ATDS2100 FPGA Physical Design System

Description

The ATDS2110PC and ATDS2120PC are complete design packages for users who want to use Viewlogic Schematic Capture and Viewsim functional and timing simulation. The ATDS2110PC provides 10K gate simulation while the ATDS2120PC provides 20K gate simulation. Both the ATDS2110 and ATDS2120 contain all the features of the ATDS2100PC and ATDS2101PC with the addition of Viewsim simulation.

ATDS2110PC and ATDS2120PC Design Flow



Circuit Verification Tools

Atmel's Circuit Verification Tools use Viewsim before and after layout to anticipate real-world circuit operating behavior. Viewlogic's Viewsim program is one of the most versatile and powerful simulators available today for use on a personal computer. Designers avoid trial and error debug by finding problems quickly and eliminating them. (Designers with a restricted ASIC/FPGA Workview license should order a Circuit Verification Upgrade Kit.)

Viewsim Functional Simulator

The Functional Simulator simulates designs in flat and hierarchical schematics representing up to 20,000 gates, and reports signal status. Entire designs can be simulated, or portions of a schematic can be isolated for analysis. Any physical node can be controlled and observed, incorporating worst-case temperature, power and process conditions.

Simulation Netlist Generator

The Simulation Netlist Generator compiles post-layout timing information in a netlist format so that the design implementation, including layout effects, is reflected in the simulation netlist.

Viewsim Full Simulator

The Full Simulator performs post-layout timing and functional verification. Viewsim's 28-state logic simulator incorporates post-layout timing results, including pin-to-pin delays, setup and hold time constraints and actual wire delays to provide accurate performance evaluations. Results can be displayed in a waveform processor to graphically measure timing relationships between signals.

Simulation Utilities

Simulation Utilities print simulation results in waveform, tabular or command language formats for use in design documentation.

Ordering Information

Ordering Code	Description
ATDS2110PC	FPGA Physical Design System with Viewdraw and 10K Gate Simulation
ATDS2110PCI	FPGA Physical Design System, Viewlogic 10K Upgrade
ATDS2120PC	FPGA Physical Design System with Viewdraw and 20K Gate Simulation
ATDS2120PCI	FPGA Physical Design System, Viewlogic 20K Upgrade

FPGA Integrated Development System Prototype Kit

Features

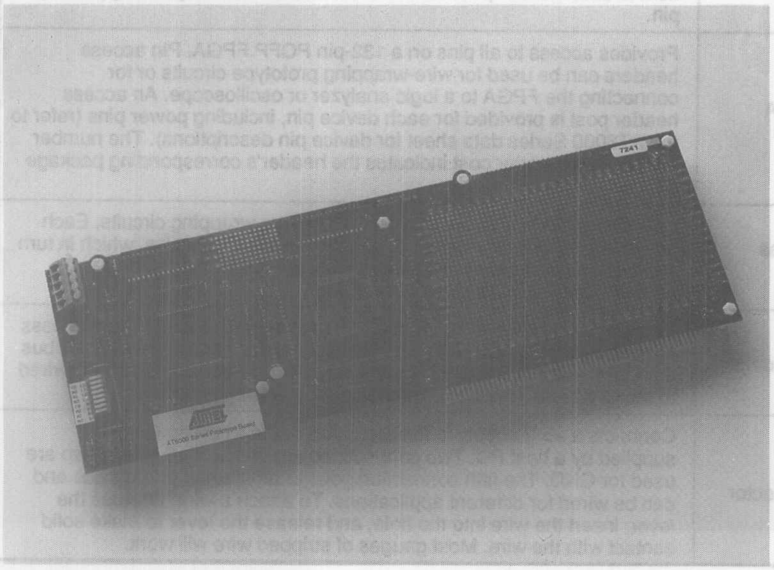
- **Prototype Board**
  - **PC/AT-Compatible Expansion Board Edge Connector**
  - **84-Pin PLCC High-Pressure Tin Socket with Solder Tails**
  - **132-Pin PQFP Micro-Pitch Socket**
  - **Over 16 Square Inches of Wire-Wrap Area**
  - **Test-Point Headers for Easy Access to Vcc, GND, and FPGA I/O Signals**
  - **Pre-Wired Header for Download Cable Connection**
  - **Pre-Wired Sockets for Serial or Parallel Configuration EPROMs**
- **Materials for Device Configuration via PC Parallel Port**
  - **Cable Adapter Nodule**
  - **DB25-Male-to-DB25-Female Cable**
  - **Ten-Conductor Ribbon Cable**

Description

Atmel's Prototype Kit lets engineers try out an FPGA design in silicon and analyze its operation in a systems environment. The Kit includes a PCB assembly with sockets for two FPGAs and space for adding active components. The board can be set up to run in at least three ways:

- Attached to a host PC running the AT6000 series design software. The Design Manager control panel is used to send configuration data to the board via the download cable supplied with the Integrated Development System.
- Attached to a PC that isn't running the Atmel Integrated Development System. The Integrated Development System generates download files that can be used on any DOS-based PC to send configuration data to the board.
- Not attached to a host PC (stand-alone). Sockets on the board are used to hold EPROMs programmed with configuration data. Power is supplied to the board from an external source.

AT6000 Series Prototype Board



Board Contents (in alphabetical order)

Symbol	Name	Description
C1-C12	Bypass Capacitors	Used to stabilize the V_{CC} and GND lines and filter out unwanted frequency components. Capacitors are placed near each FPGA to minimize any transient currents arising from device switching and magnetic coupling. Bulk capacitors placed near power supply connections accommodate the continually changing I_{CC} requirements of the total power system.
D1	Power LED	Lights up when power is applied to the board.
E1, E3	AT28C010 Parallel EPROM Socket	Holds standard 128K x 8 parallel EEPROMs (e.g., Atmel AT28C010), which can be used with configuration modes 1, 2, or 5. Table 3 lists recommended EPROMs. The configuration application note (included with the Integrated Development System) has more on configuration modes.
E2, E4	Serial EPROM Socket	Holds standard serial EPROMs (e.g., AT&T ATT1765), which can be used with configuration modes 3 or 4. Table 3 lists recommended EPROMs. The configuration application note (included with the Integrated Development System) has more on configuration modes.
F1	Power Supply Fuse (optional)	The board is wired with a zero-ohm resistor which can be replaced with a solid-state fuse or current-limiting device when such protection is desired.
J1	84-Pin PLCC FPGA Download Cable Connector	Attaches the download cable to the board so a PC can be used to configure an 84-pin device using mode 3.
J2	132-Pin PQFP FPGA Download Cable Connector	Attaches the download cable to the board so a PC can be used to configure a 132-pin device using mode 3.
J3, J4, J5, J6	84-Pin PLCC FPGA Pin Access Headers	Provides access to all pins on an 84-pin PLCC FPGA. Pin access headers can be used for wire-wrapping prototype circuits or for connecting the FPGA to a logic analyzer or oscilloscope. An access header post is provided for each device pin, including power pins (refer to the AT6000 Series data sheet for device pin descriptions). The number next to each header post indicates the header's corresponding package pin.
J7, J8, J9, J10	132-Pin PQFP FPGA Pin Access Headers	Provides access to all pins on a 132-pin PQFP FPGA. Pin access headers can be used for wire-wrapping prototype circuits or for connecting the FPGA to a logic analyzer or oscilloscope. An access header post is provided for each device pin, including power pins (refer to the AT6000 Series data sheet for device pin descriptions). The number next to each header post indicates the header's corresponding package pin.
J11, J12	Power Supply Access Headers	Provides access to power and ground for wire wrapping circuits. Each power supply access header shares a column of +5 V pins, which in turn share power and ground with the FPGA, EPROMs, configuration headers, and AT Bus edge connectors.
J13, J14	AT Bus Access Headers	Provides access to the AT edge connector signals. Each AT bus access header is connected to a finger. The power labels correspond to the bus standard. The GND and +5 V edge connector headers are already wired to the GND and +5 V power planes on the prototype board.
J15	Power Supply Connector	Connects a +5 V supply to the prototype board when power is not supplied by a host PC. Two connections are used for +5 V, and two are used for GND. The fifth connection goes to an unused board trace and can be wired for different applications. To attach a wire, depress the lever, insert the wire into the hole, and release the lever to make solid contact with the wire. Most gauges of stripped wire will work.

Board Contents (continued)

Symbol	Name	Description
L0-L15	Ground Loops	Connects a logic analyzer or oscilloscope to the ground plane for cleaner, simpler test setup.
SW1-8	Mode Select Switches SW8 M0 84-pin PLCC SW7 M1 84-pin PLCC SW6 M2 84-pin PLCC SW5 $\overline{CS}$ 84-pin PLCC SW4 M0 132-pin PQFP SW3 M1 132-pin PQFP SW2 M2 132-pin PQFP SW1 $\overline{CS}$ 132-pin PQFP	Selects the configuration mode. Each switch is connected to one of the configuration control pins (M0, M1, M2, or $\overline{CS}$). When the switch is open, it drives the input with a logic "1." When closed, it drives the input with a logic "0." The logical "1" and "0" positions are noted on the board silkscreen.
U1	84 Pin PLCC Socket AMP Socket # 821573-1 AMP Tool # 821590-1	Holds any AT6000 Series device housed in an 84-pin PLCC package. The location of power, ground, and configuration pins is constant throughout the family, making the socket compatible with any AT6000 Series device in this package. The arrow on the socket indicates Pin 1. The 84-pin PLCC can be inserted by hand, but an extraction tool from AMP Incorporated (84 PLCC #821590-1) can make removal easier.
U2	132 pin PQFP Socket AMP Socket # 821949-5 AMP Cover # 821942-1 AMP Tool # 821958-2 AMP Sheet # 15-9516	Holds any AT6000 Series device housed in a 132-pin PLCC package. The location of power, ground, and configuration pins is constant throughout the family, making the socket compatible with any AT6000 Series device in this package. The arrow on the socket indicates Pin 1. An insertion and extraction tool from AMP Incorporated (132 PQFP #821958-2) is recommended for inserting and removing the PQFP.

Board Setup and Configuration Requirements

Attaching to a PC Running the Integrated Development System

This setup lets the engineer configure devices from within the Design Manager environment. There engineer doesn't have to pull the FPGA out of the board or reprogram EPROMs for each design iteration.

Board Setup

To link the board to a host PC, use the DB25-male-to-DB25-female cable to connect the PC parallel or printer port (LPT1: or LPT2:) to the cable adapter module. Use the ten-conductor ribbon cable to connect the cable adapter module to the configuration headers J1 and J2. Figures 1 and 2 show how to connect the cable adapter module to the 84-pin and 132-pin sockets respectively.

Bit streams sent over the PC parallel port use mode 3 (bit-sequential, external CCLK—see the AT6000 Series configuration application note for more information about configuration modes). Make sure the FPGA is set up for mode 3 before applying power to the board.

To configure a device in the 84-pin socket, set mode switch SW6 to 0, SW7 to 1, and SW8 to 1. This selects the correct settings for mode 3 configuration: M2=0, M1=1, and M0=1. Close the $\overline{CS}$ switch SW5 to supply a logical "0" to the $\overline{CS}$ input.

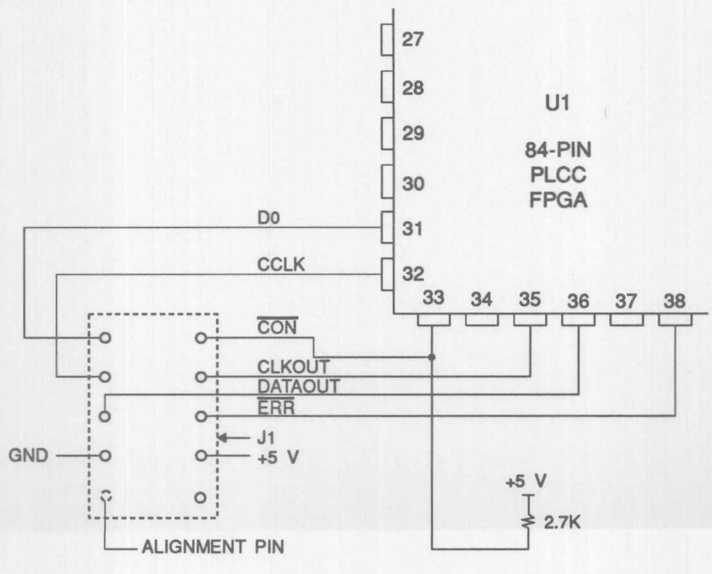
To configure a device in the 132-pin socket, set mode switch SW2 to 0, SW3 to 1, and SW4 to 1. This selects the correct settings for the mode 3 configuration: M2=0, M1=1, and M0=1. Close $\overline{CS}$ switch SW1 to supply a logical "0" to the $\overline{CS}$ input.

It is very important that the PC and the prototype board power supply be capable of sharing a common ground. The download cable connects the PC parallel port ground to the prototype board ground. The Schmitt trigger module gets its power (~5mA) from the prototype board. The user is responsible for solving any ground contention problems between the PC and the user power supply. *Atmel is not responsible for damage to the PC or power supply caused by improper grounding of the lab setup.*

Once the board is properly connected to the PC and the mode switches are in their proper position, a power supply of between 4.75 and 5.25 volts can be applied to jumper J15. It is recommended that the supply be a minimum of 1 ampere to accommodate any current spikes generated by high-speed, simultaneously switching outputs. If the power LED does not light once power is supplied, immediately disconnect power and check the polarity of hand-wired circuits and the power supply fuse for possible problems.

The prototype board can be treated like an add-in board and put inside the PC chassis. Place the board inside the PC and attach it to the PC/AT-compatible bus slot. Turn the PC power supply off and *do not* use the power supply connector J1 with this setup. The download cable should still be connected to the parallel port and the prototype board in the PC. Power is supplied from the PC bus.

Figure 1. Ribbon Cable Header Connecting 84-Pin Socket



Initiating Configuration

Instructions for using the Design Manager to generate the bit stream appear in the Integrated Development System User's Guide. When generating the bit stream, make sure of four things:

1. The bit stream is generated using mode 3.
2. The $\overline{\text{CHECK}}$ function is disabled or the $\overline{\text{CHECK}}$ pin is tied high on the prototype board.
3. The $\overline{\text{ERR}}$ pin is operating without interference so it can verify the download process.
4. LPT1: is the default printer port. To use LPT2:, change the default setting using the Design Manager's environment setup utility.

Attaching to a PC that Isn't Running the Integrated Development System

A security block restricts the Integrated Development System to a single PC, but an executable download file (provided with the Integrated Development System) makes configuration more portable. A total of four files are used to download configurations to the prototype board from any PC/AT:

- downld.exe
- <design name>.bst
- design.cfg
- <design name>.lst

These files are particularly useful if the test equipment needed to analyze the design is in a different room or if design work is divided among many engineers.

Board Setup

The prototype board connects to the host PC in the same way as described in the previous section.

PC Setup

The following Integrated Development System files are use to download a bit stream from the host PC:

- downld.exe

  Downld.exe transfers the .bst file from the PC to the prototype board. It is stored with the other executables in the \cli\bin\ directory. It is the only executable file used by the host PC.

  It uses the dual-function $\overline{\text{ERR}}$ pin to detect errors in downloading the configuration file. If $\overline{\text{ERR}}$ goes low, an error has occurred. The error is displayed on the screen and noted in a list file, downld.lst (see below). To prevent false errors, make sure the $\overline{\text{ERR}}$ pin operates without interference during the download process. Refer to the AT6000 Series configuration application note for more on dual-function pins.

- <design name>.bst

  Contains the bit stream used to configure the device. The Integrated Development System User's Guide describes how to generate a <design name>.bst file.

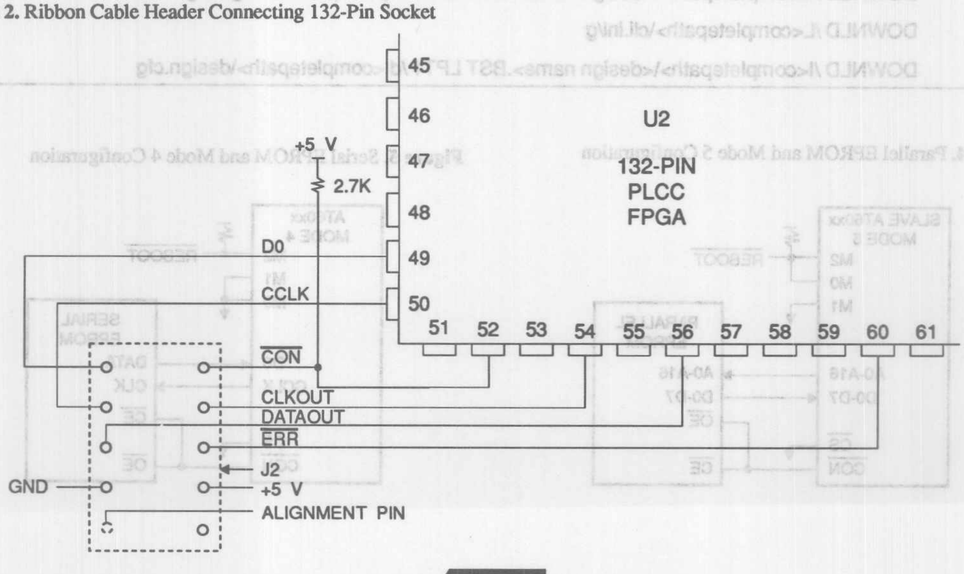
- design.cfg

  Contains the default parameters of the host PC. It is typically stored in the same directory as the <design name>.bst file.

- downld.lst

  All messages resulting from the prototype operation are directed to this file, including the time the program was initiated, the time the program completed, and the port used. It is typically stored in the same directory as the <design name>.bst file.

Figure 2. Ribbon Cable Header Connecting 132-Pin Socket



Follow these steps to set up the host PC for downloading configuration. Refer to the Integrated Development System User's Guide for more details.

1. Load the downld.exe file onto the PC.
2. Create a working directory, and a project sub-directory for data files and the design.cfg file.
3. Place a cli.ini file, edited to reflect the proper path for the project sub-directory and name, in the new working directory.
4. Place the <design name>.bst file in the project sub-directory.
5. Place the design.cfg file, edited to specify the parallel port to be used (LPT1: is the default), in the project sub-directory.

Initiating Configuration

Table 1 lists commands used to invoke the downld.exe files. Port can be either of the parallel printer ports LPT1 or LPT2. LPT1 is the default. The port argument is case-sensitive.

Configuring as a Stand-Alone (Not Attached to a Host PC)

It is possible to use the prototype board without the download cable. The prototype board is supplied with sockets for serial and parallel configuration EPROMs. Stand-alone operation can be used to test and verify EPROMs before production. It can also help in the verification of a prototype system using auto configuration. Modes 4 and 5 load configuration data from the EPROM right after the application of power. Mode 5 does not automatically generate a $\overline{\text{CON}}$ signal after the power-up boot sequence. Configuration is initiated by driving $\overline{\text{CON}}$ low. Serial EPROMs are best suited for mode 4 configuration, parallel EPROMs for mode 5.

Board Setup

Sockets E1 and E3 hold standard 64K-x-8 parallel EPROMs (e.g., AT 28C010), which can be used with configuration modes 1, 2, or 5. Figure 3 shows how the socket is wired to the FPGA.

Sockets E2 and E4 hold standard serial EPROMs (e.g., ATT1765), which can be used with configuration modes 3 or 4. Figure 4 shows how the socket is wired to the FPGA.

The AT6000 Series configuration application note gives more detailed configuration requirements. When setting up the board for auto-configuration, make sure of three things:

1. The configuration mode select switches are set to the proper value before power is applied to the board.
2. The $\overline{\text{CS}}$ switch for the device being used is closed.
3. If a mode other than 4 or 5 is to be used, a configuration clock must be supplied by the user. Configuration is initiated by pulsing the $\overline{\text{CON}}$ pin low.

In this setup, power is applied through the J15 jumper, or the board is installed into a PC bus. Make sure the mode switches are set and the programmed EPROM is properly installed before applying power to the host PC.

Table 1. Commands Used to Invoke Download.exe Files

DOWNLD <design name>.[PORT]
DOWNLD /I<completepath>\<design name>.BST /d <complete path>\design.cfg
DOWNLD /L<completepath>\cli.ini/g
DOWNLD /I<completepath>\<design name>.BST LPT1 /d <completepath>\design.cfg

Figure 4. Parallel EPROM and Mode 5 Configuration

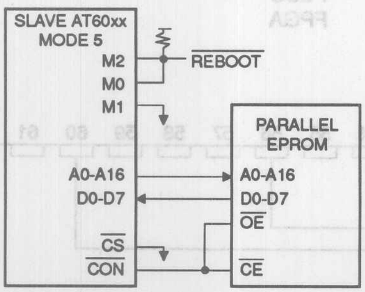
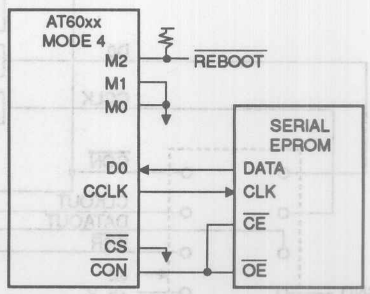


Figure 5. Serial EPROM and Mode 4 Configuration



EPROM Setup

The Integrated Development System can generate a bit stream in the Intel HEX format, accepted by most EPROM programmers. To generate a bit stream in HEX format, run the bit stream or cascade the program with the /h option:

bitstream /g /h
or
bitstream <design name> /h

See the Integrated Development System User's Guide for more information.

A Note About EPROMs and EEPROMs

Virtually any EPROM with an access time faster than 500 ns and an active-low output enable control is suitable for storing configuration data. FPGA density and required configuration speed determine which EPROM is most appropriate.

The amount of memory needed to hold configuration data increases as the FPGA gets larger. The AT6002 needs 2,676 bytes (21,408 bits) of storage capacity for full configuration; the AT6005 needs 8,077 bytes (64,616 bits). Compressed and partial-configuration bit streams use fewer bits and require less space.

Atmel and other vendors carry suitable parallel EPROMs. AT&T and Xilinx offer serial EPROMs large enough for the AT6000 series. The ones from AT&T are reprogrammable; the ones from Xilinx are not (Table 2). All these EPROMs are available in "through-hole" DIPs and surface-mount packages. They can be programmed using industry-standard programmers, like those from DATA I/O Corporation.

Table 2. Recommended EPROMs

	Part Number	Size
Parallel EEPROMs		
Atmel	AT28C64	8K x 8
	AT28C256	32K x 8
	AT28C010	128K x 8
Serial EPROMs		
AT&T	ATT1736	36K x 1
	ATT1765	65K x 1
	ATT17128	128K x 1
Xilinx	XC-1736	36K x 1
	XC-1765	65K x 1
	XC-17128	128K x 1

Contents

Section 4 **CMOS Gate Arrays**

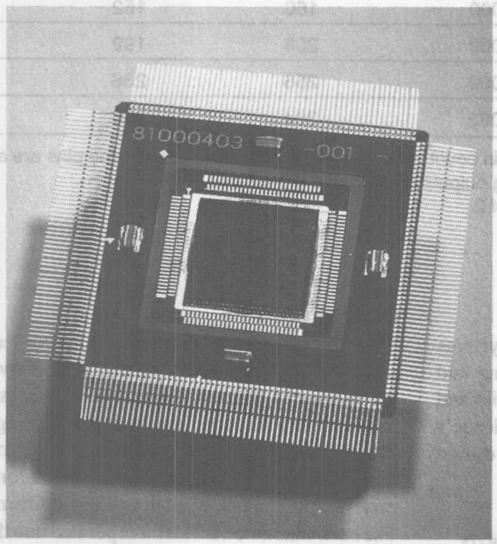
Features

- 1.0 μ Drawn Gate Length High-performance CMOS Gate Arrays
- All ATL Gate Arrays are Specified from 3.0 Volts to 5.5 Volts, for Standard and Low Voltage Applications
- Design Translation of Existing ASIC Designs Provide for Easy Alternate Sourcing with Equivalent or Improved Performance
- EPLD/FPGA Conversions to ATL Gate Array, Several EPLD and FPGA can be Combined into a Single Gate Array.
- ATL C Version, Fine Pad Pitch Gate Arrays are Ideal for High I/O, Low Gate Count Designs (Commercial, Industrial Only)
- ATL Gate Arrays can be Supplied Compliant to MIL-STD-883
- Improved Product Testability Using Serial Scan, Boundary Scan, JTAG and Built-in-self-test

Description

The high-performance ATL Series CMOS gate arrays employ 1.0 μ-drawn, double-level metal, Si-gate, CMOS technology processed in Atmel's U.S.-based, advanced manufacturing facility. The arrays utilize an enhanced channelless architecture which results in greater than 50 percent usable gates.

Atmel's flexible design system uses industry design standards and is compatible with popular CAD/CAE software and hardware packages. The customer can start designing with the ATL series today using existing CAD/CAE tools.



ATL Series Gate Arrays

ATL4
ATL10
ATL20
ATL40
ATL60
ATL75
ATL100
ATL130
ATL160

ATL7C
ATL10C
ATL15C
ATL20C
ATL35C
ATL55C
ATL75C

4

Number	Gates	Gates	Count	Pins	Speed
ATL4	4,100	2,600	68	60	375 ps
ATL10	10,000	6,500	124	116	375 ps
ATL20	22,000	12,000	144	136	375 ps
ATL40	40,000	22,000	180	168	375 ps
ATL60	57,000	30,000	224	208	375 ps
ATL75	72,000	38,000	256	236	375 ps
ATL100	95,000	50,000	292	262	375 ps
ATL130	131,000	67,000	338	308	375 ps
ATL160	157,000	80,000	360	320	375 ps

ATL C Array Organization - Fine Pad Pitch

Device Number	Maximum Gates	Routable Gates	Max Pin Count	Max I/O(1) Pins	Gate(2) Speed
ATL7C	7,000	4,000	100	92	375 ps
ATL10C	10,000	6,000	120	112	375 ps
ATL15C	15,000	8,000	144	136	375 ps
ATL20C	22,000	12,000	160	152	375 ps
ATL35C	35,000	18,000	208	192	375 ps
ATL55C	55,000	29,000	256	236	375 ps
ATL75C	75,000	39,000	304	280	375 ps

Notes: 1. Absolute maximum I/O pins is maximum pin count minus 8. Additional power and ground pins are assumed to be required to support simultaneous switching outputs as pin count increases.
2. Nominal 2 input nand gate with a fan out of 2

ATL Design

Design Systems Supported

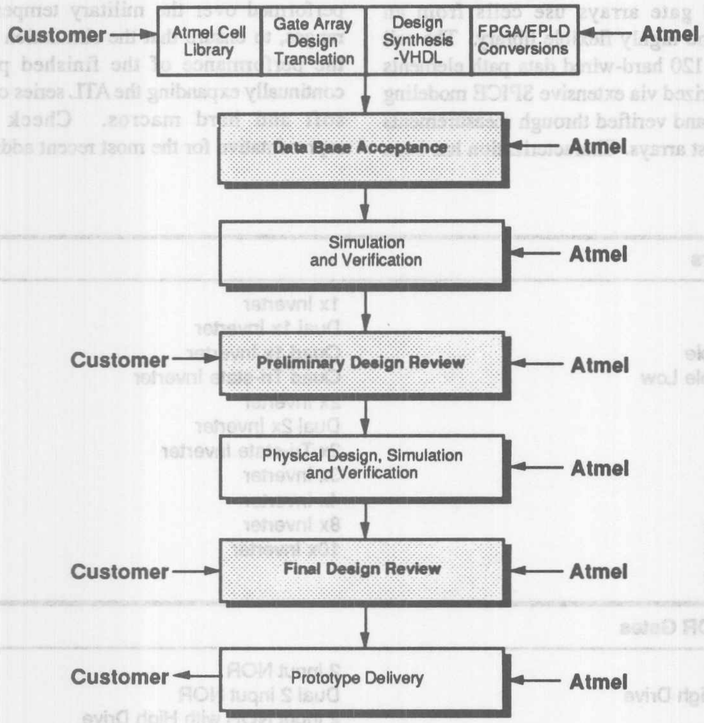
Atmel supports the major CAE/CAD software systems with complete macro cell libraries (symbols, timing and function), as well as utilities for checking the netlist and accurate pre-route delay simulations. Atmel uses Cadence's Verilog-XL as our golden simulator. The following design systems are supported:

Cadence	Viewlogic	Mentor	Dazix
Valid	Synopsys	Racal-Redac	

Design Flow

While Atmel provides four options for implementing a gate array design, they all have the same flow. Data base acceptance is the first milestone. This is when Atmel receives and accepts the complete design data base. Preliminary design review is where the performance of the design is set based on the Cadence simulation. Final design review is the last review of the design before making masks. The back annotation data is incorporated into the simulations. After final design review masks are released and prototypes in ceramic packages are delivered.

ATL Gate Array Design Flow



Design Options

Schematic Capture

Schematic capture and simulation are performed by the customer using an Atmel supplied macro cell library. The customer can also receive complete back annotation delay data for post-route simulation.

VHDL/Verilog-HDL

Atmel can accept Register Transfer level (RTL) designs for VHDL (MIL-STD-454, IEEE STD 1076) or Verilog-HDL format. Atmel fully supports Synopsys for VHDL simulation as well as synthesis. Design via VHDL or Verilog-HDL is the preferred method of performing a gate array design.

ASIC Design Translation

Atmel has successfully translated dozens of existing designs from most major ASIC vendors (LSI Logic, Oki, NEC, Fujitsu and others) into our ATL series gate arrays. These designs have been optimized for speed, gate count, modified to add logic or memory, or replicated for a pin-for-pin compatible, drop-in replacement.

FPGA and EPLD Conversions

Atmel has successfully translated existing FPGA/EPLD designs from most major vendors (Xilinx, Actel, Altera, AMD & Atmel) into our ATL series gate arrays. The design can be optimized for speed or power consumption, modified to add logic or memory or replicated for a pin-for-pin compatible, drop-in replacement. Atmel frequently combines several devices onto a single gate array.

ATL Series Cell Library

Atmel's ATL series gate arrays use cells from an accurately modeled and highly flexible library. The cell library contains over 120 hard-wired data path elements and has been characterized via extensive SPICE modeling at the transistor level and verified through measurements made on fabricated test arrays. Characterization has been performed over the military temperature and voltage ranges, to ensure that the simulation accurately predicts the performance of the finished product. Atmel is continually expanding the ATL series cell library with both soft and hard macros. Check with your sales representative for the most recent additions.

Cell Guide

Buffers and Inverters	
1x Buffer	1x Inverter
2x Buffer	Dual 1x Inverter
2x Buffer with Enable	Quad 1x Inverter
2x Buffer with Enable Low	Quad Tri-state Inverter
3x Buffer	2x Inverter
4x Buffer	Dual 2x Inverter
8x Buffer	2x Tri-state Inverter
12x Buffer	3x Inverter
16x Buffer	4x Inverter
Delay Buffer 2.0 ns	8x Inverter
Delay Buffer 3.5 ns	10x Inverter
Delay Buffer 8.0 ns	

AND, NAND, OR, NOR Gates	
2 input AND	2 input NOR
2 input AND with High Drive	Dual 2 input NOR
3 input AND	2 input NOR with High Drive
3 input AND with High Drive	3 input NOR
4 input AND	3 input NOR with High Drive
4 input AND with High Drive	4 input NOR
5 input AND	4 input NOR with High Drive
2 input NAND	5 input NOR
Dual 2-input NAND	8 input NOR
2 input NAND with High Drive	16 input NOR with High Drive
3 input NAND	2 input OR
3 input NAND with High Drive	2 input OR with High Drive
4 input NAND	3 input OR
4 input NAND with High Drive	3 input OR with High Drive
5 input NAND	4 input OR
5 input NAND with High Drive	4 input OR with High Drive
6 input NAND	
6 input NAND with High Drive	
8 input NAND	
8 input NAND with High Drive	

Cell Guide

Multiplexers	
2:1 MUX	4:1 MUX
2:1 MUX with High Drive	4:1 MUX w/o Buffered Inputs
Inverting 2:1 MUX w/o Buffered Inputs	4:1 MUX w/o Buffered Inputs, High Drive
Inverting 2:1 MUX w/o Buffered Inputs, High Drive	5:1 MUX with High Drive
2:1 MUX with Enable Low	8:1 MUX
Quad 2:1 MUX with Enable	8:1 MUX with Enable Low
Quad 2:1 MUX	8:1 MUX High Drive
Inverting 3:1 MUX w/o Buffered Inputs	
Inverting 3:1 MUX w/o Buffered Inputs, High Drive	
AND/OR, OR/AND Gates	
3 input AND OR INVERT	3 input OR AND INVERT
3 input AND OR INVERT with High Drive	3 input OR AND INVERT with High Drive
4 input AND OR INVERT	4 input OR AND INVERT
4 input AND OR INVERT with High Drive	4 input OR AND INVERT with High Drive
6 input AND OR INVERT	8 input OR AND INVERT
6 input AND OR INVERT with High Drive	4 input OR AND INVERT with 2 inputs to AND
Exclusive OR/NOR Gates	
1 bit Adder	2 input Exclusive OR with High Drive
1 bit Adder with Buffered Outputs	2 input Exclusive NOR
7 input Carry Lookahead	2 input Exclusive NOR with High Drive
2 input Exclusive OR	
Decoders	
2:4 Decoder	3:8 Decoder with Low Enable
2:4 Decoder with Low Enable	
Flip-flops/Latches	
D Flip-flop	LATCH
D Flip-flop with Clear/Preset	LATCH with Complementary Outputs
D Flip-flop with Clear	LATCH with Inverted Gate Signal
D Flip-flop with High Drive	QUAD LATBG with Common Gate Signal
D Flip-flop with Reset	LATCH with High Drive
D Flip-flop with Set	QUAD Inverting LATCH
D Flip-flop with Set/Reset	LATCH with Reset
JK Flip-flop	LATCH with Set
JK Flip-flop with Clear/Preset	LATCH with Set and Reset
JK Flip-flop with Clear	

Cell Guide

Scan Cells	
Set-scan Register	Set-scan Register with Set
Set-scan Register with Clear and Preset	Set-scan Register with Set and Reset
Set-scan Register with Reset	

I/O Options

Input, Output, Bidirectional, Tristate Output, Internal Clock Driver and Oscillator

Output Drive Value Programmable from 2 mA to 24 mA in 2 mA increments with Slew Rate Control

CMOS or TTL Operation

Schmitt Trigger (Bidirectional, Input)

Testable NAND Gate on Input (Bidirectional, Input)

Inverting and Non-inverting Input Buffers (Bidirectional, Input)

Pullup Resistor - 10K Ω to 310K Ω

Pulldown Resistor - 3.5K Ω to 108.5K Ω

74XX Series Soft Macros

24 cells available

HDL Macros - Available in Verilog-HDL or VHDL Simulation Models

Function Group	Available Cells
adder	37
alu	29
baud rate generator	3
comparator	18
counter	27
fifo	56
incrementor/decrementor	60
mux	7
parity/error correction	15
scan	31
shifter	9
multipliers	10

CMOS/TTL Input Interface Characteristics

Interface	Logic High	Logic Low	Switchpoint
CMOS	3.5 V Minimum	1.5 V Maximum	V_{dd}/2 Typical
TTL	2.0 V Minimum	0.8 V Maximum	1.4 V Typical

Absolute Maximum Ratings*

Operating Temperature	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-2.0 V to +7.0 V[1]
Maximum Operating Voltage	6.0 V

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Notes:
1. Minimum voltage is -0.6 V dc which may undershoot to -2.0 V for pulses of less than 20 ns. Maximum output pin voltage is V_{dd} + 0.75 V dc which may overshoot to +7.0 V for pulses of less than 20 ns.

5.0 Volt DC Characteristics

Applicable over recommended operating range from T_a = -55°C to +125°C, V_{dd} = 4.5 V to 5.5 V (unless otherwise noted)

Symbol	Parameter	Test Condition	Min	Typ	Max	Units
I_{IH}	Input Leakage High	$V_{IN} = V_{dd}$, V_{dd} = 5.5 V		.01	10	µA
I_{IL}	Input Leakage Low (no pull-up)	$V_{IN} = V_{SS}$, V_{dd} = 5.5 V	-10	.01		µA
	40K pull-up	$V_{IN} = V_{SS}$, V_{dd} = 5.5 V	-325	-160	-40	µA
I_{OZ}	Output Leakage (no pull-up)	$V_{IN} = V_{dd}$ or V_{SS}, V_{dd} = 5.5 V	-10	.01	10	µA
I_{OS}	Output Short Circuit Current (3 x Buffer)(2)	V_{dd} = 5.5 V, $V_{OUT} = V_{dd}$	10	50	100	mA
		V_{dd} = 5.5 V, $V_{OUT} = V_{SS}$	-100	-50	-10	mA
V_{IL}	TTL Input Low Voltage				0.8	V
V_{IL}	CMOS Input Low Voltage				0.3 x V_{dd}	V
V_{IH}	TTL Input High Voltage		2.0			V
V_{IH}	CMOS Input High Voltage		0.7 x V_{dd}			V
V_T	TTL Switching Threshold	V_{dd} = 5.0 V, 25°C		1.4		V
	CMOS Switching Threshold	V_{dd} = 5.0 V, 25°C		2.4		V
V_{OL}	Output Low Voltage Output buffer has 12 stages of drive capability with 2 mA I_{OL} per stage.	I_{OL} = as rated V_{dd} = 4.5 V		0.2	0.4	V
V_{OH}	Output High Voltage Output buffer has 12 stages of drive capability with -2 mA I_{OH} per stage.	I_{OH} = as rated V_{dd} = 4.5 V	0.7 x V_{dd}	4.2		V

Note: 2. This is the specification for the 3 x Output Buffer. Output short circuit current for other outputs will scale accordingly. Not more than one output shorted at a time, for a maximum of one second, is allowed.

3.3 Volt DC Characteristics

Applicable over recommended operating range from T_a = -55°C to +125°C, V_{dd} = 3.0 V to 3.6 V (unless otherwise noted)

Symbol	Parameter	Test Condition	Min	Typ	Max	Units
I_{IH}	Input Leakage High	$V_{IN} = V_{dd}$, V_{dd} = 3.6 V		.01	10	µA
I_{IL}	Input Leakage Low (no pull-up)	$V_{IN} = V_{SS}$, V_{dd} = 3.6 V	-10	.01		µA
	40K pull-up	$V_{IN} = V_{SS}$, V_{dd} = 3.6 V	-200	-60	-10	µA
I_{OZ}	Output Leakage (no pull-up)	$V_{IN} = V_{dd}$ or V_{SS}, V_{dd} = 3.6 V	-10	.01	10	µA
I_{OS}	Output Short Circuit Current	V_{dd} = 3.6 V, $V_{OUT} = V_{dd}$	5	25	60	mA
	(3 x Buffer)(2)	V_{dd} = 3.6 V, $V_{OUT} = V_{SS}$	-60	-25	-5	mA
V_{IL}	CMOS Input Low Voltage				0.3 x V_{dd}	V
V_{IH}	CMOS Input High Voltage		0.7 x V_{dd}			V
V_T	CMOS Switching Threshold	V_{dd} = 3.3 V, 25°C		1.5		V
V_{OL}	Output Low Voltage Output buffer has 12 stages of drive capability with 1 mA I_{OL} per stage.	I_{OL} = as rated V_{dd} = 3.0 V			0.4	V
V_{OH}	Output High Voltage Output buffer has 12 stages of drive capability with -1 mA I_{OH} per stage.	I_{OH} = as rated V_{dd} = 3.0 V	0.7 x V_{dd}			V

Note: 2. This is the specification for the 3 x Output Buffer. Output short circuit current for other outputs will scale accordingly. Not more than one output shorted at a time, for a maximum of one second, is allowed.

I/O Buffer DC Characteristics

Symbol	Parameter	Test Condition	Min	Typ	Max	Units
C_{IN}	Capacitance Input Buffer (Die)	5.0 V, 3.3 V		2.4		pF
C_{OUT}	Capacitance Output Buffer (Die)	5.0 V, 3.3 V		5.6		pF
$C_{I/O}$	Capacitance Bi-Directional	5.0 V, 3.3 V		6.6		pF
Schmitt Trigger						
V+	TTL Positive Threshold	25°C, 5.0 V		1.8	2.0	V
	CMOS Positive Threshold	25°C, 5.0 V		3.2	3.5	V
V-	TTL Negative Threshold	25°C, 5.0 V	0.6	0.8		V
	CMOS Negative Threshold	25°C, 5.0 V	1.0	1.2		V
ΔV	TTL Hysteresis	25°C, 5.0 V	0.4	1.0		
	CMOS Hysteresis	25°C, 5.0 V	1.0	2.0		
V+	CMOS Positive Threshold	25°C, 3.3 V		2.2	2.3	V
V-	CMOS Negative Threshold	25°C, 3.3 V	.65	0.9		V
ΔV	CMOS Hystersis	25°C, 3.3 V	.65	1.3		

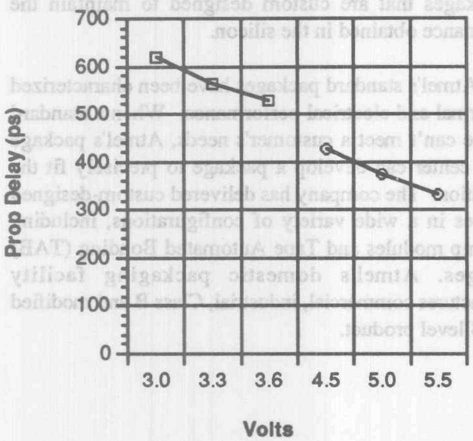
I/O Buffers

- Programmable output drive (2 to 24 mA I_{OL}, -2 to -24 mA I_{OH} for 5.0 V 1 to 12 mA I_{OL}, -1 to -12 mA I_{OH} for 3.3 V)
- 3000 volts ESD protection
- Built-in configurable test logic

The ATL series input/output ring contains the I/O buffer circuitry capable of sourcing and sinking currents up to 24 mA, and responds to CMOS or TTL logic levels. I/O locations on this ring can accommodate bidirectional cells.

AC Characteristics

Delay vs V_{dd}

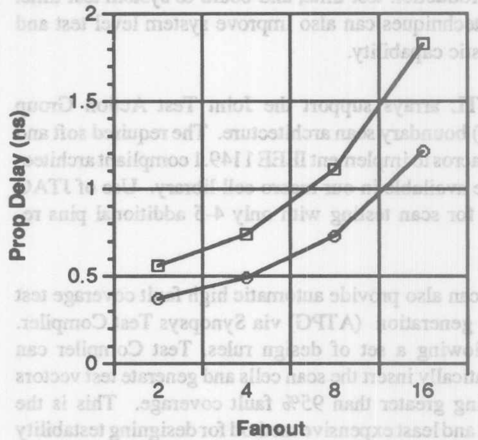


3.3 Volts V_{dd}

5.0 Volts V_{dd}

NAND2 - 2 input NAND
Temp = 25°C
FO = 2

Delay vs Fanout

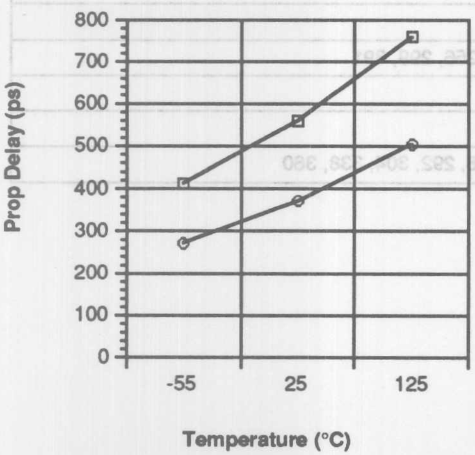


3.3 Volts V_{dd}

5.0 Volts V_{dd}

NAND2 - 2 input NAND
Temp = 25°C

Delay vs Temperature

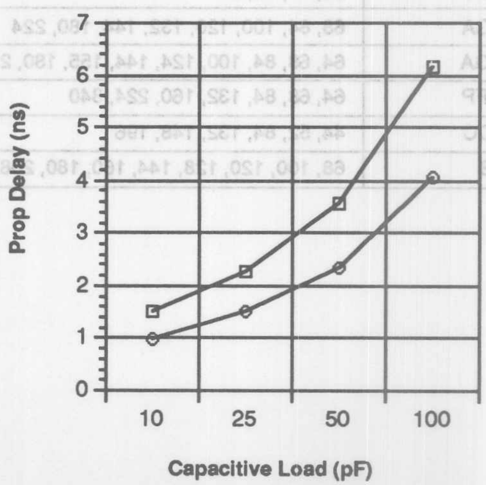


3.3 Volts V_{dd}

5.0 Volts V_{dd}

NAND2 - 2 input NAND
FO = 2

Output Buffer vs Load



3.3 Volts V_{dd}

5.0 Volts V_{dd}

PDO4 - Output Buffer 8 mA
Temp = 25°C

4

Design for Testability

Atmel supports a full range of Design-for-Test improvement techniques which reduce design and prototype debug time, production test time, and board & system test time. These techniques can also improve system level test and diagnostic capability.

The ATL arrays support the Joint Test Action Group (JTAG) boundary scan architecture. The required soft and hard macros to implement IEEE 1149.1 compliant architecture are available in our macro cell library. Use of JTAG allows for scan testing with only 4-5 additional pins required.

Atmel can also provide automatic high fault coverage test pattern generation (ATPG) via Synopsys Test Compiler. By following a set of design rules, Test Compiler can automatically insert the scan cells and generate test vectors providing greater than 95% fault coverage. This is the easiest and least expensive method for designing testability into a gate array design.

Advanced Packaging

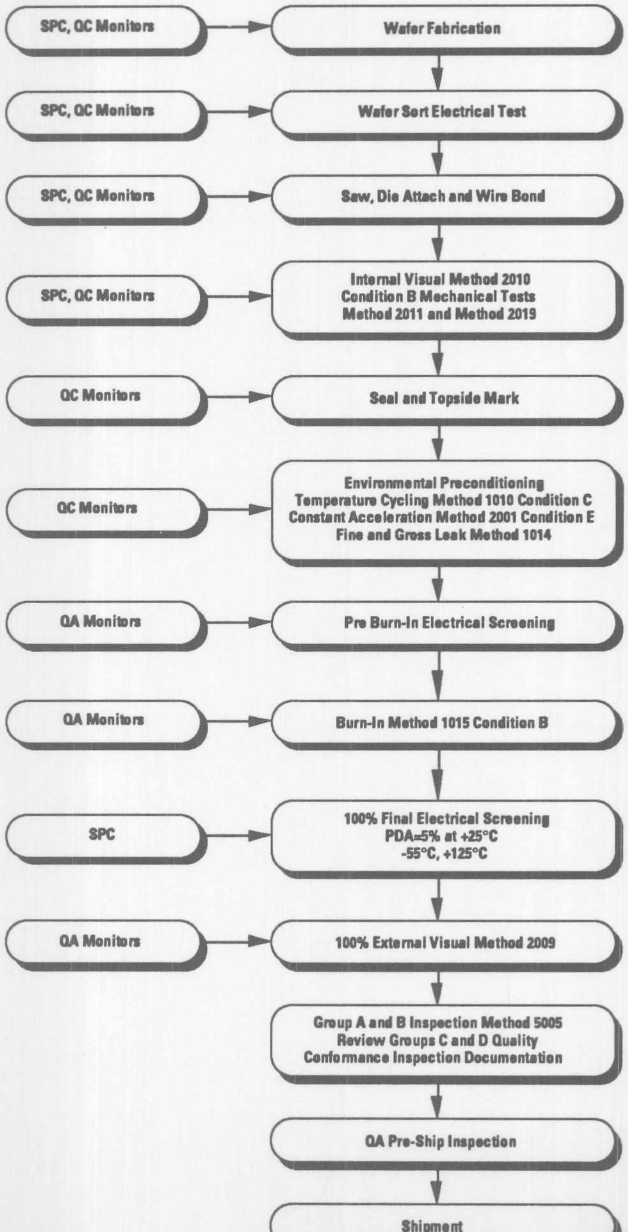
Atmel supports a wide variety of standard packages for the ATL series, but also offers its ATL series gate arrays in packages that are custom designed to maintain the performance obtained in the silicon.

All of Atmel's standard packages have been characterized for thermal and electrical performance. When a standard package can't meet a customer's needs, Atmel's package design center can develop a package to precisely fit the application. The company has delivered custom-designed packages in a wide variety of configurations, including multichip modules and Tape Automated Bonding (TAB) packages. Atmel's domestic packaging facility manufactures commercial, industrial, Class B and modified Class S level product.

Packaging Options

Package Type	Pin Count
TQFP	44, 48, 64, 80, 100, 144, 160, 208, 240, 248, 304
PQFP	44, 64, 68, 80, 100, 120, 128, 132, 136, 144, 160, 184, 208, 232, 256, 304
PLCC	28, 44, 68, 84
PPGA	68, 84, 100, 120, 132, 144, 180, 224
CPGA	64, 68, 84, 100, 124, 144, 155, 180, 223, 224, 256, 299, 391
CQFP	64, 68, 84, 132, 160, 224, 340
CLCC	44, 52, 84, 132, 148, 196
TAB	68, 100, 120, 128, 144, 160, 180, 208, 224, 256, 292, 304, 338, 360

Military Product Flow Chart
MIL-STD-883 Class B



Actel, Altera, AMD, Cadence, DAZIX, Fujitsu, LSI Logic, Mentor, NEC, Oki, Racal-Redac, Synopsys, Valid, Verilog-XL, Viewlogic, and Xilinx may be registered trademarks of others.

Features

- 0.8 μ drawn gate length combined with triple level metal provides outstanding speed/density performance.
- All ATL80 arrays can operate at 5.0 volts and 3.3 volts for low-power applications. The ATL80 series can also operate in a mixed voltage environment.
- Design translation of existing ASIC, FPGA and PLD designs provide for easy alternate sourcing with equivalent performance.
- Product testability is improved using techniques such as serial and boundary scan, ATPG, built-in self test and JTAG.
- ATL80 arrays can be screened to MIL-STD-883.

ATL80 Series Gate Arrays 0.8 Micron

ATL80/15
ATL80/25
ATL80/40
ATL80/50
ATL80/75
ATL80/95
ATL80/150
ATL80/220
ATL80/280
ATL80/350
ATL80/450
ATL80/600

Description

The high-performance ATL80 Series CMOS gate arrays offer superior system performance, flexibility, testability and board utilization. The ATL80 gate arrays employ an advanced technology 0.8 μ-drawn, triple-level metal, Si-gate, CMOS technology processed in a U.S.-based, manufacturing facility.

Atmel's efficient routing scheme combined with tight spacing for three metal layers allows Atmel to provide more gates and faster speeds. With fine pitch bond pads as a standard feature, high I/O gate arrays can easily be accommodated. The ATL80 gate array can have a 3.3 volt or 5.0 volt core, combined with a 3.3 volt and/or 5.0 volt I/O on the same chip. Atmel's I/O can be personalized to accept a 5.0 volt input signal into a 3.3 volt buffer.

Atmel's flexible design system uses industry design standards and is compatible with popular CAD/CAE software and hardware packages. The 0.8 μ macro cell libraries are upward compatible with the existing 1.0 μ libraries and design utilities. The customer can start designing with the ATL80 series today using existing CAD/CAE tools.

ATL80 Array Organization

Device Number	Raw Gates	Routable Gates	Max Pin Count	Max I/O Pins	Gate(1) Speed
ATL80/15	17,000	10,200	100	92	256 ps
ATL80/25	26,000	15,600	120	112	256 ps
ATL80/40	39,000	23,400	144	136	256 ps
ATL80/50	50,000	30,000	160	152	256 ps
ATL80/75	75,000	45,000	184	176	256 ps
ATL80/95	94,000	60,000	208	192	256 ps
ATL80/150	150,000	75,000	256	236	256 ps
ATL80/220	220,000	110,000	304	280	256 ps
ATL80/280	280,000	140,000	340	310	256 ps
ATL80/350	350,000	175,000	380	350	256 ps
ATL80/450	450,000	225,000	424	384	256 ps
ATL80/600	600,000	300,000	480	440	256 ps

Note: 1. Nominal 2 Input Nand Gate With a Fan Out of 2

ATL80 Design

Design Systems Supported

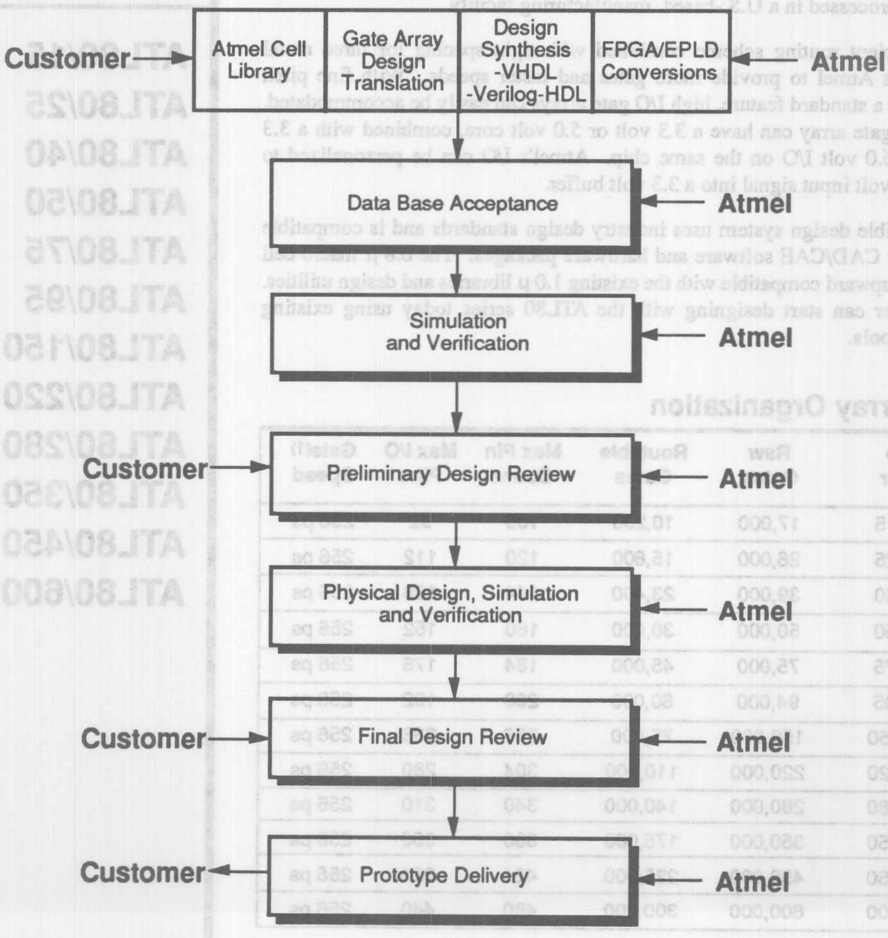
The ATL80 gate arrays are supported on the same design systems as our 1.0µ ATL gate arrays. Atmel supports the major CAE/CAD software systems with complete macro cell libraries (symbols, timing and function), as well as utilities for checking the netlist and accurate pre-route delay simulations. Atmel uses Cadence's Verilog-XL as our golden simulator. The following design systems are supported:

Cadence/Composer	Viewlogic
Cadence/Concept	Synopsys
Mentor	Dazix

Design Flow

The design flow for the ATL80 gate arrays is the same as the 1.0µ ATL gate arrays. While Atmel provides four options for implementing a gate array design, they all have the same flow. Data base acceptance is the first milestone. This occurs when Atmel receives and accepts the complete design data base. Preliminary design review occurs when the performance of the design is set based on the Cadence simulation. Final design review is the last review of the design before making masks. The back annotation data is incorporated into the simulations. After final design review masks are released and prototypes, in ceramic packages, are delivered.

ATL80 Gate Array Design Flow



Design Options

Schematic Capture

Schematic capture and simulation are performed by the customer using an Atmel supplied macro cell library. The customer can also receive complete back annotation delay data for post-route simulation.

VHDL/Verilog-HDL

Atmel can accept Register Transfer Level (RTL) designs for VHDL (MIL-STD-454, IEEE STD 1076) or Verilog-HDL format. Atmel fully supports Synopsys for VHDL simulation as well as synthesis. Design via VHDL or Verilog-HDL is the preferred method of performing a gate array design.

ASIC Design Translation

Atmel has successfully translated dozens of existing designs from most major ASIC vendors (LSI Logic, Oki, NEC, Fujitsu, AMI and others) into our gate arrays. These designs have been optimized for speed, gate count, modified to add logic or memory, or replicated for a pin-for-pin compatible, drop-in replacement.

FPGA and EPLD Conversions

Atmel has successfully translated existing FPGA/EPLD designs from most major vendors (Xilinx, Actel, Altera, AMD & Atmel) into our gate arrays. The design can be optimized for speed or power consumption, modified to add logic or memory or replicated for a pin-for-pin compatible, drop-in replacement. Atmel frequently combines several devices onto a single gate array.

ATL80 Series Cell Library

Atmel's ATL80 series gate arrays use cells from an accurately modeled and highly flexible library. The cell library contains over 120 hard-wired data path elements and has been characterized via extensive SPICE modeling at the transistor level and verified through measurements made on fabricated test arrays. Characterization has been performed over the military temperature and voltage ranges to ensure that the simulation accurately predicts the performance of the finished product. Atmel is continually expanding the ATL80 series cell library with both soft and hard macros. Check with your sales representative for the most recent additions.

Cell Guide

Buffers and Inverters	
1x Buffer	1x Inverter
2x Buffer	Dual 1x Inverter
2x Buffer with Enable	Quad 1x Inverter
2x Buffer with Enable Low	Quad Tristate Inverter
3x Buffer	2x Inverter
4x Buffer	Dual 2x Inverter
8x Buffer	2x Tri-state Inverter
12x Buffer	3x Inverter
16x Buffer	4x Inverter
Delay Buffer 1.5 ns	8x Inverter
Delay Buffer 2.1 ns	10x Inverter
Delay Buffer 6.0 ns	

AND, NAND, OR, NOR Gates	
2 input AND	2 input NOR
2 input AND with High Drive	Dual 2 input NOR
3 input AND	2 input NOR with High Drive
3 input AND with High Drive	3 input NOR
4 input AND	3 input NOR with High Drive
4 input AND with High Drive	4 input NOR
5 input AND	4 input NOR with High Drive
2 input NAND	5 input NOR
Dual 2-input NAND	8 input NOR
2 input NAND with High Drive	16 input NOR with High Drive
3 input NAND	2 input OR
3 input NAND with High Drive	2 input OR with High Drive
4 input NAND	3 input OR
4 input NAND with High Drive	3 input OR with High Drive
5 input NAND	4 input OR
5 input NAND with High Drive	4 input OR with High Drive
6 input NAND	5 input OR
6 input NAND with High Drive	
8 input NAND	
8 input NAND with High Drive	

Cell Guide

Multiplexers	
2:1 MUX	4:1 MUX
2:1 MUX with High Drive	4:1 MUX w/o Buffered Inputs
Inverting 2:1 MUX w/o Buffered Inputs	4:1 MUX w/o Buffered Inputs, High Drive
Inverting 2:1 MUX w/o Buffered Inputs, High Drive	5:1 MUX with High Drive
2:1 MUX with Enable Low	8:1 MUX
Quad 2:1 MUX with Enable Low	8:1 MUX with Enable Low
Quad 2:1 MUX	8:1 MUX High Drive
Inverting 3:1 MUX w/o Buffered Inputs	
Inverting 3:1 MUX w/o Buffered Inputs, High Drive	
AND/OR, OR/AND Gates	
3 input AND OR INVERT	3 input OR AND INVERT
3 input AND OR INVERT with High Drive	3 input OR AND INVERT with High Drive
4 input AND OR INVERT	4 input OR AND INVERT
4 input AND OR INVERT with High Drive	4 input OR AND INVERT with High Drive
6 input AND OR INVERT	8 input OR AND INVERT
6 input AND OR INVERT with High Drive	4 input OR AND INVERT with 2 inputs to AND
Exclusive OR/NOR Gates	
1 bit Adder	2 input Exclusive OR with High Drive
1 bit Adder with Buffered Outputs	2 input Exclusive NOR
7 input Carry Lookahead	2 input Exclusive NOR with High Drive
2 input Exclusive OR	
Decoders	
2:4 Decoder	3:8 Decoder with Enable Low
2:4 Decoder with Enble Low	
Flip-flops/Latches	
D Flip-flop	LATCH
D Flip-flop with Clear/Preset	LATCH with Complementary Outputs
D Flip-flop with Clear	LATCH with Inverted Gate Signal
D Flip-flop with High Drive	QUAD LATBG with Common Gate Signal
D Flip-flop with Reset	LATCH with High Drive
D Flip-flop with Set	QUAD Inverting LATCH
D Flip-flop with Set/Reset	LATCH with Reset
JK Flip-flop	LATCH with Set
JK Flip-flop with Clear	LATCH with Set and Reset

Cell Guide

Scan Cells	
Set-scan Flip-flop	Set-scan Flip-flop with Reset

I/O Options
Input, Output, Bidirectional, Tristate Output, Internal Clock Driver and Oscillator
Output Drive Value Programmable from 2 mA to 24 mA in 2 mA increments with Slew Rate Control
CMOS or TTL Operation
Schmitt Trigger (Bidirectional, Input)
Testable NAND Gate on Input (Bidirectional, Input)
Inverting and Non-inverting Input Buffers (Bidirectional, Input)
Pullup Resistor - 10K Ω to 310K Ω
Pulldown Resistor - 3.5K Ω to 108.5K Ω

74XX Series Soft Macros
24 cells available

HDL Macros - Available in Verilog-HDL or VHDL Simulation Models

Function Group	Available Cells
adder	37
alu	29
baud rate generator	3
comparator	18
counter	27
fifo	56
incrementor/decrementor	60
mux	7
parity/error correction	15
scan	31
shifter	9
multipliers	10

CMOS/TTL Input Interface Characteristics

Interface	Logic High	Logic Low	Switchpoint
CMOS	3.5 V Minimum	1.5 V Maximum	V_{DD}/2 Typical
TTL	2.0 V Minimum	0.8 V Maximum	1.4 V Typical

Absolute Maximum Ratings*

Operating Temperature	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-2.0 V to +7.0 V(1)
Maximum Operating Voltage	6.0 V

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Notes:
1. Minimum voltage is -0.6 V dc which may undershoot to -2.0 V for pulses of less than 20 ns. Maximum output pin voltage is V_{DD} + 0.75 V dc which may overshoot to +7.0 V for pulses of less than 20 ns.

5.0 Volt DC Characteristics

Applicable over recommended operating range from T_a = -55°C to +125°C, V_{DD} = 4.5 V to 5.5 V (unless otherwise noted)

Symbol	Parameter	Test Condition	Min	Typ	Max	Units
I_{IH}	Input Leakage High	$V_{IN} = V_{DD}$, V_{DD} = 5.5 V		0.01	10	µA
I_{IL}	Input Leakage Low (no pull-up)	$V_{IN} = V_{SS}$, V_{DD} = 5.5 V	-10	0.01		µA
	40K pull-up	$V_{IN} = V_{SS}$, V_{DD} = 5.5 V	-325	-160	-40	µA
I_{OZ}	Output Leakage (no pull-up)	$V_{IN} = V_{DD}$ or V_{SS}, V_{DD} = 5.5 V	-10	0.01	10	µA
I_{OS}	Output Short Circuit Current (3 x Buffer)(2)	V_{DD} = 5.5 V, $V_{OUT} = V_{DD}$	10	50	100	mA
		V_{DD} = 5.5 V, $V_{OUT} = V_{SS}$	-100	-50	-10	mA
V_{IL}	TTL Input Low Voltage				0.8	V
V_{IL}	CMOS Input Low Voltage				0.3 x V_{DD}	V
V_{IH}	TTL Input High Voltage		2.0			V
V_{IH}	CMOS Input High Voltage		0.7 x V_{DD}			V
V_T	TTL Switching Threshold	V_{DD} = 5.0 V, 25°C		1.4		V
	CMOS Switching Threshold	V_{DD} = 5.0 V, 25°C		2.4		V
V_{OL}	Output Low Voltage Output buffer has 12 stages of drive capability with 2 mA I_{OL} per stage.	I_{OL} = as rated V_{DD} = 4.5 V		0.2	0.4	V
V_{OH}	Output High Voltage Output buffer has 12 stages of drive capability with -2 mA I_{OH} per stage.	I_{OH} = as rated V_{DD} = 4.5 V	0.7 x V_{DD}	4.2		V

Note: 2. This is the specification for the 3 x Output Buffer. Output short circuit current for other outputs will scale accordingly. Not more than one output shorted at a time, for a maximum of one second, is allowed.

3.3 Volt DC Characteristics

Applicable over recommended operating range from T_a = -55°C to +125°C, V_{DD} = 3.0 V to 3.6 V (unless otherwise noted)

Symbol	Parameter	Test Condition	Min	Typ	Max	Units
I_{IH}	Input Leakage High	$V_{IN} = V_{DD}$, V_{DD} = 3.6 V		0.01	10	μA
I_{IL}	Input Leakage Low (no pull-up)	$V_{IN} = V_{SS}$, V_{DD} = 3.6 V	-10	0.01		μA
	40K pull-up	$V_{IN} = V_{SS}$, V_{DD} = 3.6 V	-200	-60	-10	μA
I_{OZ}	Output Leakage (no pull-up)	$V_{IN} = V_{DD}$ or V_{SS}, V_{DD} = 3.6 V	-10	0.01	10	μA
I_{OS}	Output Short Circuit Current (3 x Buffer)(2)	V_{DD} = 3.6 V, $V_{OUT} = V_{DD}$ V_{DD} = 3.6 V, $V_{OUT} = V_{SS}$	5 -60	25 -25	60 -5	mA mA
V_{IL}	CMOS Input Low Voltage				0.3 x V_{DD}	V
V_{IH}	CMOS Input High Voltage		0.7 x V_{DD}			V
V_T	CMOS Switching Threshold	V_{DD} = 3.3 V, 25°C		1.5		V
V_{OL}	Output Low Voltage Output buffer has 12 stages of drive capability with 1 mA I_{OL} per stage.	I_{OL} = as rated V_{DD} = 3.0 V			0.4	V
V_{OH}	Output High Voltage Output buffer has 12 stages of drive capability with -1 mA I_{OH} per stage.	I_{OH} = as rated V_{DD} = 3.0 V	0.7 x V_{DD}			V

Note: 2. This is the specification for the 3 x Output Buffer. Output short circuit current for other outputs will scale accordingly. Not more than one output shorted at a time, for a maximum of one second, is allowed.

I/O Buffer DC Characteristics

Symbol	Parameter	Test Condition	Min	Typ	Max	Units
C_{IN}	Capacitance, Input Buffer (Die)	5.0 V, 3.3 V		2.4		pF
C_{OUT}	Capacitance, Output Buffer (Die)	5.0 V, 3.3 V		5.6		pF
$C_{I/O}$	Capacitance, Bi-Directional	5.0 V, 3.3 V		6.6		pF
Schmitt Trigger						
V+	TTL Positive Threshold CMOS Positive Threshold	25°C, 5.0 V 25°C, 5.0 V		1.8 3.2	2.0 3.5	V V
V-	TTL Negative Threshold CMOS Negative Threshold	25°C, 5.0 V 25°C, 5.0 V	0.6 1.0	0.8 1.2		V V
ΔV	TTL Hysteresis CMOS Hysteresis	25°C, 5.0 V 25°C, 5.0 V	0.4 1.0	1.0 2.0		
V+	CMOS Positive Threshold	25°C, 3.3 V		2.2	2.3	V
V-	CMOS Negative Threshold	25°C, 3.3 V	0.65	0.9		V
ΔV	CMOS Hysteresis	25°C, 3.3 V	0.65	1.3		

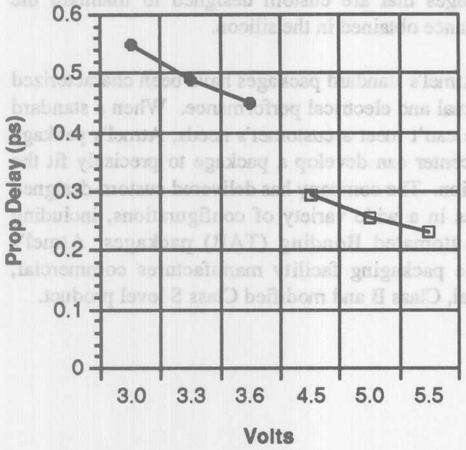
I/O Buffers

- Programmable output drive (2 to 24 mA I_{OL}, -2 to -24 mA I_{OH} for 5.0 V 1 to 12 mA I_{OL}, -1 to -12 mA I_{OH} for 3.3 V)
- 3,000 volts ESD protection
- Built-in configurable test logic

The ATL80 series input/output ring contains the I/O buffer circuitry capable of sourcing and sinking currents up to 24 mA, and responds to CMOS or TTL logic levels. All outputs can be switched to a high impedance state. I/O locations on this ring can accommodate bidirectional cells.

AC Characteristics

Delay vs V_{CC}

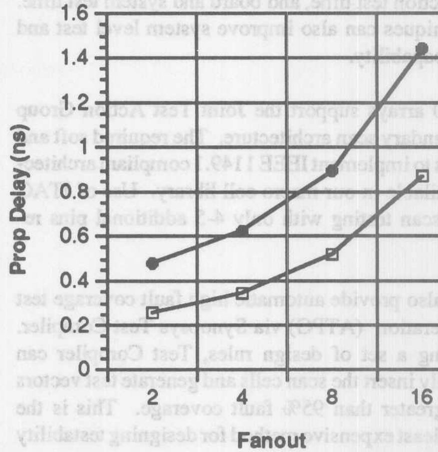


3.3 Volts V_{dd}

5.0 Volts V_{dd}

2 input NAND
Temp = 25°C
FO = 2

Delay vs Fanout

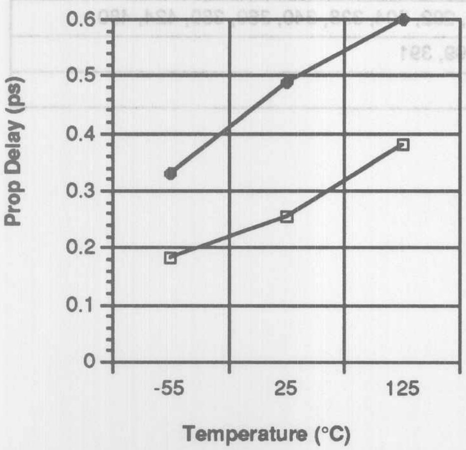


3.3 Volts V_{dd}

5.0 Volts V_{dd}

2 input NAND
Temp = 25°C

Delay vs Temperature

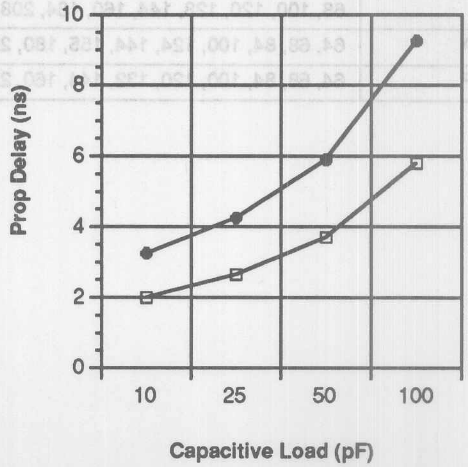


3.3 Volts V_{dd}

5.0 Volts V_{dd}

2 input NAND
FO = 2

Output Buffer vs Load



3.3 Volts V_{dd}

5.0 Volts V_{dd}

PDO4 - Output Buffer 8 mA
Temp = 25°C

Design for Testability

Atmel supports a full range of Design-for-Test improvement techniques which reduce design and prototype debug time, production test time, and board and system test time. These techniques can also improve system level test and diagnostic capability.

The ATL80 arrays support the Joint Test Action Group (JTAG) boundary scan architecture. The required soft and hard macros to implement IEEE 1149.1 compliant architecture are available in our macro cell library. Use of JTAG allows for scan testing with only 4-5 additional pins required.

Atmel can also provide automatic high fault coverage test pattern generation (ATPG) via Synopsys Test Compiler. By following a set of design rules, Test Compiler can automatically insert the scan cells and generate test vectors providing greater than 95% fault coverage. This is the easiest and least expensive method for designing testability into a gate array design.

Advanced Packaging

Atmel supports a wide variety of standard packages for the ATL80 series, but also offers its ATL80 series gate arrays in packages that are custom designed to maintain the performance obtained in the silicon.

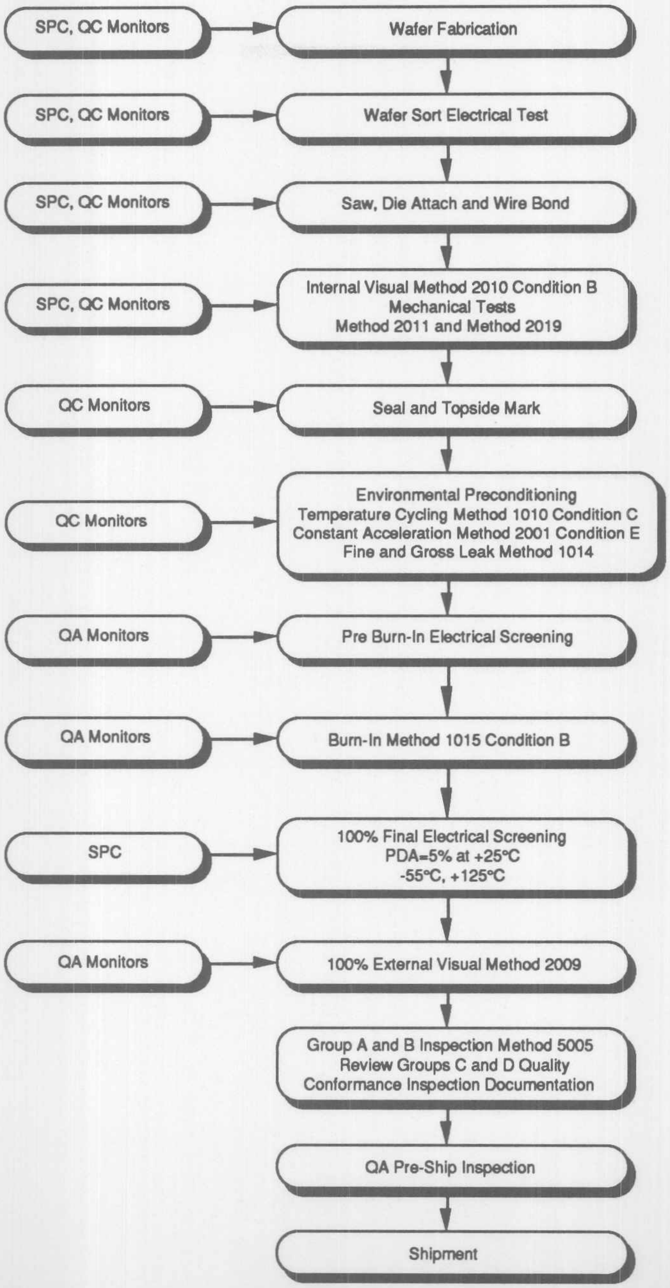
All of Atmel's standard packages have been characterized for thermal and electrical performance. When a standard package can't meet a customer's needs, Atmel's package design center can develop a package to precisely fit the application. The company has delivered custom-designed packages in a wide variety of configurations, including Tape Automated Bonding (TAB) packages. Atmel's domestic packaging facility manufactures commercial, industrial, Class B and modified Class S level product.

Packaging Options

Package Type	Pin Count
PQFP	44, 52, 64, 80, 100, 120, 128, 132, 144, 160, 184, 208, 240, 304
TQFP	44, 48, 52, 64, 80, 100, 120, 128, 144, 160, 176, 216
PLCC	20, 28, 32, 44, 52, 68, 84
TAB	68, 100, 120, 128, 144, 160, 184, 208, 224, 256, 292, 304, 338, 340, 360, 380, 424, 480
CPGA	64, 68, 84, 100, 124, 144, 155, 180, 223, 224, 299, 391
CQFP	64, 68, 84, 100, 120, 132, 144, 160, 224, 340

Military Product Flow Chart

MIL-STD-883 Class B



Features

- **Specifically Designed for Battery Powered Applications 1.0 - 3.0 Volts and will Operate from 0.7 to 5.5 Volts**
- **Static Current Drain of <75 nA at 1.0 Volts**
- **200 MHz Maximum Toggle Frequency for Flip Flop at 1.5 Volts**
- **1.0 μ Drawn Gate Length CMOS Gate Arrays**
- **All Package Styles Offered Including TQFP and TAB**
- **Improved Product Testability Using Serial Scan, Boundary Scan, and JTAG**
- **Second Source Existing ASIC Design in Atmel's ATLV via Design Translation. Improved Performance and Lower Cost**

ATLV Series Ultra Low Voltage Gate Arrays

ATLV2
ATLV3
ATLV5
ATLV7
ATLV10
ATLV15
ATLV20
ATLV35

Description

The ATLV Series CMOS gate arrays employ 1.0 μ-drawn, double-level metal, Si-gate, CMOS technology processed in Atmel's U.S.-based, advanced manufacturing facility. The arrays utilize an enhanced channelless architecture which results in greater than 50 percent usable gates.

Atmel's flexible design system uses industry design standards and is compatible with popular CAD/CAE software and hardware packages. The customer can start designing with the ATLV series today using existing CAD/CAE tools.

ATLV Array Organization

Device Number	Raw Gates	Routable Gates	Max Pin Count	Max I/O(1) Pins	Gate(2) Speed
ATLV2	2,000	1,400	44	36	1.3 ns
ATLV3	3,000	1,600	68	60	1.3 ns
ATLV5	5,000	2,800	84	76	1.3 ns
ATLV7	7,000	4,400	100	92	1.3 ns
ATLV10	10,000	6,600	120	112	1.3 ns
ATLV15	15,000	8,000	144	136	1.3 ns
ATLV20	22,000	12,000	160	152	1.3 ns
ATLV35	35,000	18,000	208	192	1.3 ns

Notes: 1. Absolute maximum I/O pins is maximum pin count minus 8. Additional power and ground pins are assumed to be required to support simultaneous switching outputs as pin count increases.
2. Nominal 2 input nand gate with a fan out of 2 at 1.5 volts, room temperature.

ATLV Design

Design Systems Supported

Atmel supports the major CAE/CAD software systems with complete macro cell libraries (symbols, timing and function), as well as utilities for checking the netlist and accurate pre-route delay simulations. Atmel uses Cadence's Verilog-XL as our golden simulator. Design systems which are supported include Cadence, Viewlogic, Mentor, Dazix and Synopsys.

Design Flow

While Atmel provides four options for implementing a gate array design, they all have the same basic flow. Data base acceptance is the first milestone. This is when Atmel receives and accepts the complete design data base. Preliminary design review is where the performance of the design is set based on the Cadence simulation. Final design review is the last review of the design before making masks. The back annotation data is incorporated into the simulations. After final design review masks are released and prototypes in ceramic packages are delivered.

Design Options

Schematic Capture

Schematic capture and simulation are performed by the customer using an Atmel supplied macro cell library. The customer can also receive complete back annotation delay data for post-route simulation.

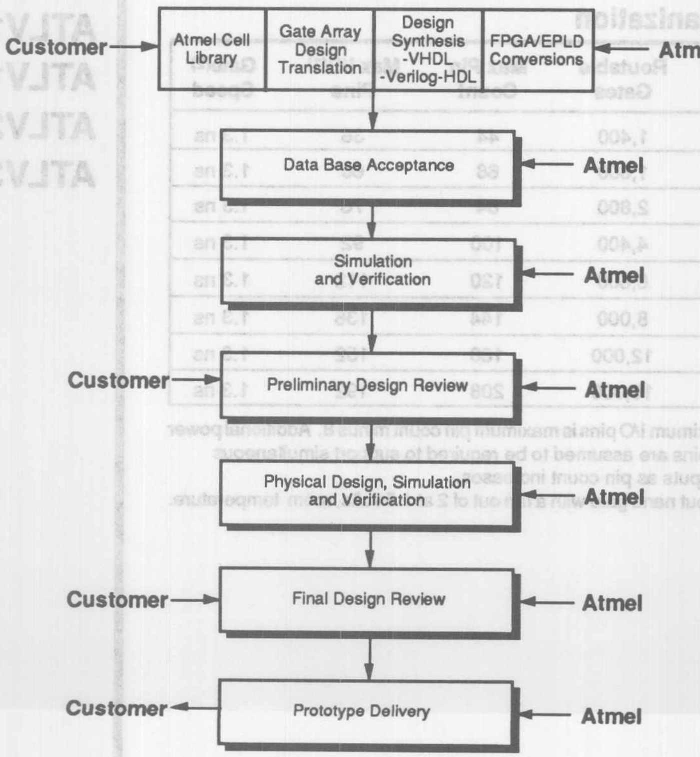
VHDL/Verilog-HDL

Atmel can accept Register Transfer level (RTL) designs for VHDL (MIL-STD-454, IEEE STD 1076) or Verilog-HDL format. Atmel fully supports Synopsys for VHDL simulation as well as synthesis. Design via VHDL or Verilog-HDL is the preferred method of performing a gate array design.

ASIC Design Translation

Atmel has successfully translated dozens of existing designs from most major ASIC vendors (LSI Logic, Oki, NEC, Fujitsu and others) into our gate arrays. These designs have been optimized for speed, gate count, modified to add logic or memory, or replicated for a pin-for-pin compatible, drop-in replacement.

ATLV Gate Array Design Flow



FPGA and EPLD Conversions

Atmel has successfully translated existing FPGA/EPLD designs from most major vendors (Xilinx, Actel, Altera, AMD & Atmel) into our gate arrays. The design can be optimized for speed or power consumption, modified to add logic or memory or replicated for a pin-for-pin compatible, drop-in replacement. Atmel frequently combines several devices onto a single gate array.

ATLV Series Cell Library

Atmel's ATLV series gate arrays use cells from an accurately modeled and highly flexible library. The cell library contains over 120 hard-wired data path elements and has been characterized via extensive SPICE modeling at the transistor level and verified through measurements made on fabricated test arrays. The symbols for the ATLV cell library are compatible with Atmel's ATL (1.0 μ 3.3 and 5.0 V) and ATL80 (0.8 μ 3.3 and 5.0 V) cell libraries. Existing designs can be easily migrated to the ATLV series. Characterization has been performed over commercial temperature and 1.0 to 3.0 volts, to ensure that the simulation accurately predicts the performance of the finished product. Atmel is continually expanding the ATLV series cell library with both soft and hard macros. Check with your sales representative for the most recent additions.

Cell Guide

Buffers and Inverters	
1x Buffer	1x Inverter
2x Buffer	Dual 1x Inverter
2x Buffer with Enable	Quad 1x Inverter
2x Buffer with Enable Low	Quad Tri-state Inverter
3x Buffer	2x Inverter
4x Buffer	Dual 2x Inverter
8x Buffer	2x Tri-state Inverter
12x Buffer	3x Inverter
16x Buffer	4x Inverter
Delay Buffer 2.0 ns	8x Inverter
Delay Buffer 3.5 ns	10x Inverter
Delay Buffer 8.0 ns	

AND, NAND, OR, NOR Gates	
2 input AND	2 input NOR
3 input AND	Dual 2 input NOR
4 input AND	3 input NOR
5 input AND	4 input NOR
2 input NAND	5 input NOR
Dual 2-input NAND	8 input NOR
3 input NAND	2 input OR
4 input NAND	3 input OR
5 input NAND	4 input OR
6 input NAND	
8 input NAND	

Multiplexers	
2:1 MUX	4:1 MUX
Inverting 2:1 MUX w/o Buffered Inputs	4:1 MUX w/o Buffered Inputs
Inverting 2:1 MUX w/o Buffered Inputs	4:1 MUX w/o Buffered Inputs
2:1 MUX with Enable Low	8:1 MUX
Quad 2:1 MUX with Enable	8:1 MUX with Enable Low
Quad 2:1 MUX	
Inverting 3:1 MUX w/o Buffered Inputs	
Inverting 3:1 MUX w/o Buffered Inputs	

Cell Guide

AND/OR, OR/AND Gates	
3 input AND OR INVERT 4 input AND OR INVERT 6 input AND OR INVERT	3 input OR AND INVERT 4 input OR AND INVERT 8 input OR AND INVERT
Exclusive OR/NOR Gates	
1 bit Adder 1 bit Adder with Buffered Outputs 7 input Carry Lookahead	2 input Exclusive OR 2 input Exclusive NOR
Decoders	
2:4 Decoder 2:4 Decoder with Low Enable	3:8 Decoder with Low Enable
Flip-flops/Latches	
D Flip-flop D Flip-flop with Clear/Preset D Flip-flop with Clear D Flip-flop with Reset D Flip-flop with Set D Flip-flop with Set/Reset JK Flip-flop JK Flip-flop with Clear/Preset JK Flip-flop with Clear	LATCH LATCH with Complementary Outputs LATCH with Inverted Gate Signal QUAD LATBG with Common Gate Signal QUAD Inverting LATCH LATCH with Reset LATCH with Set LATCH with Set and Reset
Scan Cells	
Set-scan Register Set-scan Register with Clear and Preset Set-scan Register with Reset	Set-scan Register with Set Set-scan Register with Set and Reset
I/O Options	
Input, Output, Bidirectional, Tristate Output, Internal Clock Driver and Oscillator	
Output Drive Value Programmable from 0.5 mA to 6 mA in 0.5 mA increments with Slew Rate Control	
CMOS Operation	
Testable NAND Gate on Input (Bidirectional, Input)	
Inverting and Non-inverting Input Buffers (Bidirectional, Input)	
Pullup Resistor - 10K Ω to 310K Ω	
Pulldown Resistor - 3.5K Ω to 108.5K Ω	

CMOS Input Interface Characteristics

Interface	Logic High	Logic Low	Switchpoint
CMOS	0.90 V_{DD}	0.1 V_{DD}	V_{DD}/2 Typical

Absolute Maximum Ratings*

Operating Temperature	-40°C to +85°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-2.0 V to +5.5 V[1]
Maximum Operating Voltage	5.5 V

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Notes:
1. Minimum voltage is -0.6 V dc which may undershoot to -2.0 V for pulses of less than 20 ns. Maximum output pin voltage is V_{DD} + 0.75V dc which may overshoot to +7.0 V for pulses of less than 20 ns.

1.5 Volt DC Characteristics

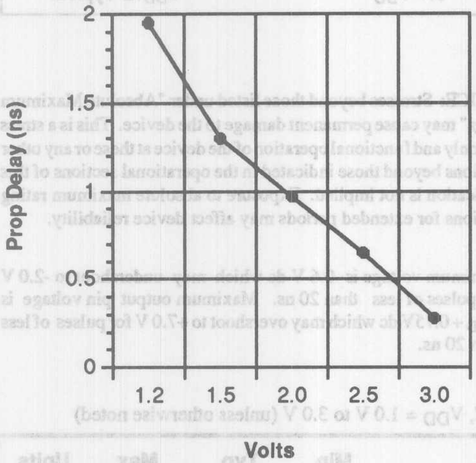
Applicable over recommended operating range from T_a = -40°C to +85°C, V_{DD} = 1.0 V to 3.0 V (unless otherwise noted)

Symbol	Parameter	Test Condition	Min	Typ	Max	Units
I_{IH}	Input Leakage High	V_{IN}=V_{DD}, V_{DD}=1.8 V		1 x 10^{-5}	10	µA
I_{IL}	Input Leakage Low (no pull-up)	V_{IN}=V_{SS}, V_{DD}=1.8 V	-10	-1 x 10^{-5}		µA
I_{OZ}	Output Leakage (no pull-up)	V_{IN}=V_{DD} or V_{SS}, V_{DD}=3.6 V	-10	1 x 10^{-5}	10	µA
I_{OS}	Output Short Circuit Current (3 x Buffer)(2)	V_{DD}=1.8 V, V_{OUT}=V_{DD} V_{DD}=1.8 V, V_{OUT}=V_{SS}	5 -60	25 -25	60 -5	mA mA
V_{IL}	CMOS Input Low Voltage				0.2 x V_{DD}	V
V_{IH}	CMOS Input High Voltage		0.8 x V_{DD}			V
V_T	CMOS Switching Threshold	V_{DD}=1.5 V, 25°C		0.75		V
V_{OL}	Output Low Voltage Output buffer has 12 stages of drive capability with 0.5 mA I_{OL} per stage.	I_{OL}=as rated V_{DD}=1.5 V			0.2 x V_{DD}	V
V_{OH}	Output High Voltage Output buffer has 12 stages of drive capability with -0.5 mA I_{OH} per stage.	I_{OH}=as rated V_{DD}=1.5 V	0.8 x V_{DD}			V
I_{DD}	Static Current Input Leakage Low (no pull-up)	1.0 V 3.0 V		< 75 < 1.0		nA µA

Note: 2. This is the specification for the 3 x Output Buffer. Output short circuit current for other outputs will scale accordingly. Not more than one output shorted at a time, for a maximum of one second, is allowed.

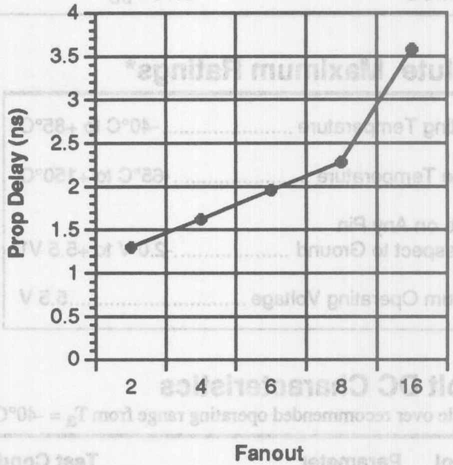
AC Characteristics

Delay vs V_{DD}



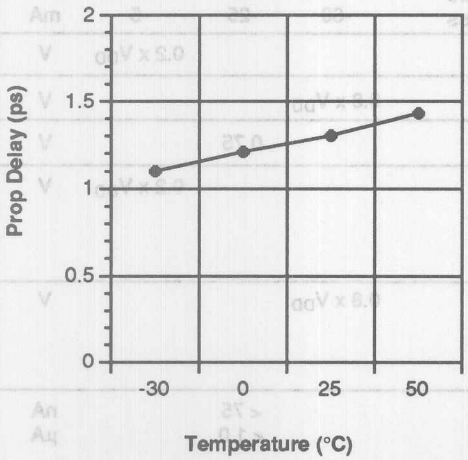
NAND2 - 2 input NAND
Temp = 25°C
FO = 2

Delay vs Fanout



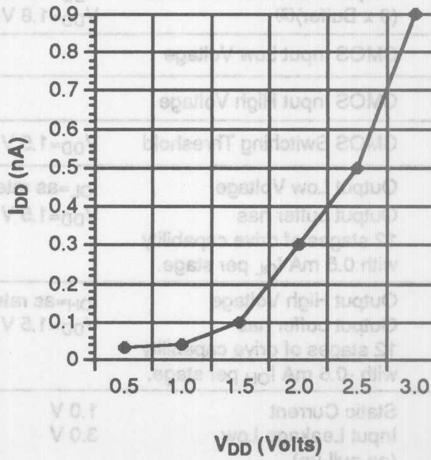
1.5 Volts V_{DD}
NAND2 - 2 input NAND
Temp = 25°C

Delay vs Temperature



1.5 Volts V_{DD}
NAND2 - 2 input NAND
FO = 2

Current Drain vs Voltage



Temp = 25°C

I/O Buffer DC Characteristics

Symbol	Parameter	Test Condition	Min	Typ	Max	Units
C_{IN}	Capacitance Input Buffer (Die)	1.5 V		2.4		pF
C_{OUT}	Capacitance Output Buffer (Die)	1.5 V		5.6		pF
$C_{I/O}$	Capacitance Bi-Directional	1.5 V		6.6		pF

I/O Buffers

- Programmable output drive
  0.5 to 6 mA I_{OL}, -4.5 to -6 mA I_{OH} for 1.5 V)
- 3000 volts ESD protection

The ATLV series input/output ring contains the I/O buffer circuitry capable of sourcing and sinking currents up to 6 mA, and responds to CMOS logic levels. I/O locations on this ring can accommodate bidirectional cells.

Design for Testability

Atmel supports a full range of Design-for-Test improvement techniques which reduce design and prototype debug time, production test time, and board & system test time. These techniques can also improve system level test and diagnostic capability.

The ATLV arrays support the Joint Test Action Group (JTAG) boundary scan architecture. The required soft and hard macros to implement IEEE 1149.1 compliant architecture are available in our macro cell library. Use of JTAG allows for scan testing with only 4-5 additional pins required.

Atmel can also provide automatic high fault coverage test pattern generation (ATPG) via Synopsys Test Compiler.

By following a set of design rules, Test Compiler can automatically insert the scan cells and generate test vectors providing greater than 95% fault coverage. This is the easiest and least expensive method for designing testability into a gate array design.

Advanced Packaging

Atmel supports a wide variety of standard packages for the ATLV series, but also offers its ATLV series gate arrays in packages that are custom designed to maintain the performance obtained in the silicon. All of Atmel's standard packages have been characterized for thermal and electrical performance.

When a standard package can't meet a customer's needs, Atmel's package design center can develop a package to precisely fit the application. The company has delivered custom-designed packages in a wide variety of configurations, including Tape Automated Bonding (TAB) packages. Atmel's domestic packaging facility manufactures commercial, industrial and Class B.

Packaging Options

Package Type	Pin Count
PQFP	44, 52, 64, 80, 100, 120, 128, 132, 144, 160, 184, 208, 240, 304
TQFP	44, 48, 52, 64, 80, 100, 120, 128, 144, 160, 176, 216
PLCC	20, 28, 32, 44, 52, 68, 84
TAB	68, 100, 120, 128, 144, 160, 184, 208, 224, 256, 292, 304, 338, 340, 360, 380, 424, 480
CPGA	64, 68, 84, 100, 124, 144, 155, 180, 223, 224, 299, 391
CQFP	64, 68, 84, 100, 120, 132, 144, 160, 224, 340

Introduction

Atmel provides one of the most flexible ASIC design systems in the industry. The company recognizes that customers develop gate arrays in a variety of different methodologies. Atmel has four different techniques for design interface:

Design with an Atmel Cell Library on a broad range of industry standard platforms and CAE/CAD Tools

Conversion of existing FPGA/PLD designs into Atmel gate arrays for improved performance or drop-in replacements

Translation of existing Gate Arrays/Standard Cells into Atmel gate arrays for improved performance or drop-in replacements

Design Synthesis via VHDL or Verilog-HDL

Many customers perform schematic capture and simulation using an Atmel supplied macro cell library. Atmel supports most popular third party design tools with complete cell libraries. Atmel can also translate existing gate array/standard cell designs and deliver parts that are pin-for-pin compatible drop-in replacements. Atmel can support the customer with optimization, improved performance (speed), or increased logic and memory. Many customers now prototype using programmable logic (FPGA/PLD). Atmel can convert Actel™, Altera™, Atmel and Xilinx™ FPGAs/PLDs into gate arrays with equivalent or improved performance. And finally Atmel can accept RTL (Register Transfer Level) designs in VHDL (MIL-STD-454L, IEEE-STD-1076) or Verilog-HDL™ for design synthesis to a gate level description.

Gate Array Design

Atmel Design Tools

System	Version	Simulator & Design Entry	Netlist Checker	Test Vector Checker	Delay Calculator & Back Annotation	Computer
Cadence/Composer™	1.3 1.6 4.2	Veritime™ Verilog-XL™ Composer™	v3	tvc	Yes	Sun 4, HP
Cadence/Concept™	10.3	Concept/GED	v3	tvc	No	Sun 4
Viewlogic™	4.2-PC 5.1-Sun	ViewSIM™, ViewDRAW™ ViewSIM™, ViewDRAW™	- v3	tvc tvc	Yes Yes	386/486 PC Sun 4
Mentor™	7.0	QuickSIM™ Neted™	v3	tvc	Yes	DN3000/4500
	8.2	QuickSIM II™ Design Architect™	v3	tvc	Yes	Sun 4, HP
Synopsys™	3.0	Design Compiler™ Test Compiler™ VHDL Simulation		tvc	Yes	Sun 4, HP

Design Flow

Atmel's design flow has four major milestones independent of the design interface used:

Database Acceptance
Preliminary Design Review
Final Design Review
Prototype Delivery

Atmel has defined specific requirements for each milestone that must be accomplished prior to moving on in the design flow.

Database Acceptance (DA)

At this milestone Atmel formally accepts the design database as complete and begins work on the design. At DA Atmel will verify that the complete database has been received, there are no known errors (netlist checker and test vector checker) and that the netlist, vectors, etc. are all in the correct format.

It is critical that all the required data elements are submitted at one time in the correct format and that the database is final (no anticipated changes).

Database Summary
Flat netlist
Test Vectors (TV) per Atmel's test vector checklist

Documentation
- v3 Report from Netlist Checker
- Report from tvc (Test Vector Checker)
- Timing Diagram (clocks, vectors applied, vectors sampled)
- Critical paths
- System load
- Purpose of TV sets

A detailed database acceptance checklist for each of the four design flows (cell library, ASIC translation, FPGA/PLD conversion and VHDL/Verilog-HDL) provides the designer with the exact requirements.

Preliminary Design Review (PDR)

After DA Atmel will migrate all designs into the Cadence Design System. Atmel uses Cadence's Verilog-XL/Veritime as our golden simulator and all designs are signed off based on the Cadence results. The submitted design is ported to Verilog and re-simulated using the customer supplied vectors. Functional performance is verified as well as key timing performance. Once this has been successfully completed, a PDR is held to agree upon the performance level of the gate array. Following are the requirements for PDR:

Confirm v3 file correct
Confirm tvc file correct
I/O buffer listing and bonding diagram
Preliminary testability compiler report indicating areas that may not be covered
Route clock tree and analysis of worst case and best case delay
Verilog simulation - 1 speed
Nominal, worst case, best case (with no hold time violations)
- review clock timing
- at-speed
- clock skew (if required)
- listing of timing warnings with explanation
Review critical path information (t_{SU}, t_{HOLD}, t_{PD})
- Verilog or Veritime estimates
I/O electrical
Electromigration calculation pre-FDR

ASIC Translation:
Minicell List
- cells used
- Atmel mappings
- notation for new cells and new mappings

I/O buffer list with same information as minicell list
- I/O buffer Atmel comparisons
Timing comparison for all cells
New soft macro mappings
Document netlist changes due to mapping

Physical Design

Once the circuit has been fully simulated and evaluated for compliance to the specification and performance goals, the netlist is translated into the Cadence Gate Ensemble™ physical design tool (previously known as Tangate™). After place-and-route, the design is resimulated to incorporate the effects of parasitic capacitance networks associated with the double and triple metal routing of the physical design (back annotation data). These effects will already have been anticipated in the initial simulations by the customer and by Atmel since a typical parasitic capacitance loading will have been supplied with the cell library timing models. Therefore, performance changes between pre- and post-route simulation results are minimal.

Basic place-and-route is a routine production operation that allows very sophisticated parts (over 300,000 usable gates) to be routed in a batch mode in a short period of time. The tools guarantee 100 percent routing with an approach that is amenable to fast design cycle times.

The Cadence delay calculation model includes effects such as: intrinsic delay, transition time, extra source delay, parasitic capacitance, and effects of various loading conditions. Timing assurance supports timing-driven layout based on either path or net constraints. Path constraints allows the customer or Atmel to identify critical paths, typically done using static path timing analysis, and then placing a constraint on Cadence to meet boundary value delays. With the net constraints approach, the boundary values are specified in units of time or capacitance for any given net in the netlist. The placement algorithm simultaneously considers routability, power balancing and timing requirements.

Back Annotation or Post-Route Simulation

Cadence does an extraction of timing related data - either by creating a delay report or an RC report. The delay report allows for an immediate static-path timing analysis, while the RC report allows data to be exported back to the baseline simulator, Verilog-XL , to confirm that the previously incorporated estimates for parasitic RC associated with placement and routing were accurate. The timing performance of the circuit is updated at this time with the customer prior to going to silicon fabrication. The post-route simulation is done with customer supplied load conditions and with the ATE (automatic test equipment) load conditions so that a test specification can also be signed off at FDR. A back annotation report will be provided to the customer in the supported design systems. This will give the customer the most accurate timing information possible.

4

Design Verification

Part of Atmel's flexible design system is the industry standard for design verification - DRACULA™. Dracula has options for:

ERC - electrical rule checking
DRC - design rule checking
LVS - layout versus schematic comparison

The base array and cell library have already been through this checkout procedure and have been verified in silicon using test vehicles containing cells from the Atmel libraries. This process and design characterization data resulted in the tabularized information used by the proprietary delay calculator described earlier. It is necessary to verify that the personalization levels accurately reflect both the customer functional circuit and the CMOS process design rules before masks are made. The results of these verification steps are also available at the FDR for customer sign-off. Together with the extensive timing analysis, verification guarantees to the customer that prototype parts will meet the specification.

Final Design Review (FDR)

The Final Design Review is the last joint review between Atmel and the customer before committing to prototypes. Prior to the meeting both Atmel and the customer will have reviewed the post-route Verilog-XL simulation incorporating the back annotation data, and the customer will have completed post-route simulation on their CAE systems. Atmel guarantees silicon performance equal to or better than predicted on the post-route Verilog-XL simulations. Below are the requirements for FDR.

Confirm v3 and tvc files correct
-post-route netlist changes
I/O buffer listing and bonding diagram
Testability report
Clock tree and analysis of worst case and best case delay
Verilog simulation - with back annotation data
Nominal, worst case, best case (with no hold time violations)
-review clock timing
-at-speed
-clock skew (if required)
-listing of timing warnings with explanation
Review critical path information (t_{SU}, t_{HOLD}, t_{PD})
-Verilog or Veritime results
I/O Electrical
Electromigration calculation
Ground bounce analysis
LVS/DRC/ERC

Prototype Delivery

Atmel will deliver 10 prototypes in ceramic packages to the customer. The purpose of the units is to verify the design's functionality and electrical performance.

Cell Library

The traditional approach to designing ASICs is for the customer to perform schematic capture of the logic diagram using a CAE workstation that has the ASIC supplier's symbols and timing libraries resident on the computer (see Figure 3-1). The customer is required to perform schematic entry, test vector generation and functional simulation. Atmel uses Verilog-XL™ as our golden simulator, however, other simulators are also supported.

Many problems with simulation, verification and test can be prevented by incorporating good testability techniques up front. In particular, generation of a sufficient number of vectors to achieve a desired level of fault grading (95%) can be extremely time consuming if proper design techniques are not used.

Atmel Cell Library Design Flow

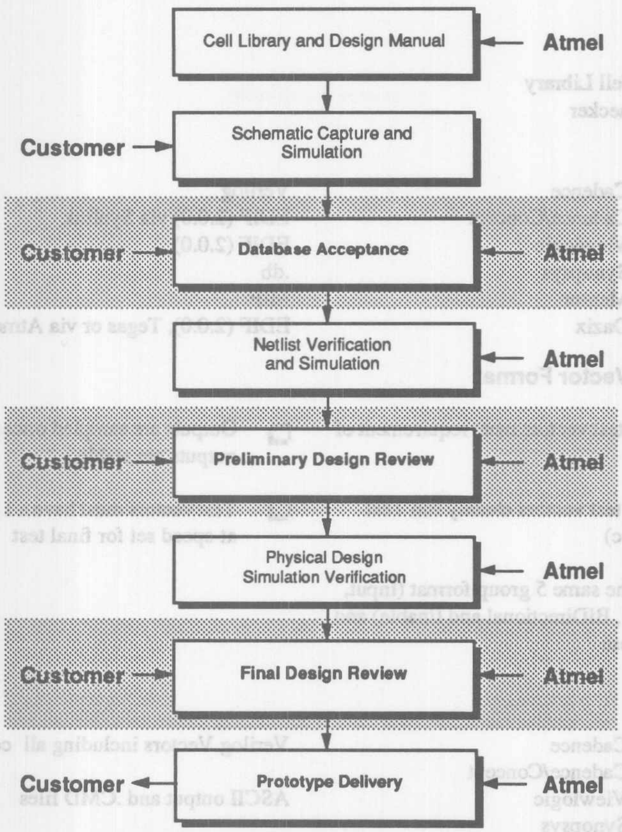


Figure 3-1

Atmel Cell Library Database Acceptance Check List

Database

Netlist Format

Flatten netlist in Atmel Cell Library
Cleanly run v3 Netlist Checker

Acceptable Formats:

Cadence	Verilog
Cadence/Concept	EDIF (2.0.0) via Synlink
Viewlogic	EDIF (2.0.0)
Synopsys	.db
Mentor	MIF
Dazix	EDIF (2.0.0), Tegas or via Atmel Dazix Netlister

Simulation and Test Vector Format

- ☐ All simulation and test vectors meet requirement of Tester Check List
- ☐ All simulation and test vectors cleanly run Test Vector Checker (tvc)
- ☐ All vectors are in the same 5 group format (Input, Output, Tri State™, BiDirectional and Enable) and have a stated purpose
- ☐ Outputs are sampled once per clock cycle when the outputs are stable.
- ☐ Test vectors must have a 1 MHz set for probe and an at-speed set for final test

Acceptable Formats:

Cadence	Verilog Vectors including all control files
Cadence/Concept	
Viewlogic	ASCII output and .CMD files
Synopsys	
Mentor	QuickSIM .LOG, .LIST and .FORCE files including .DO files
Dazix	ASCII Vectors and time stamp

Specifications

- Operating conditions; Temperature and voltage, best nominal, worst (case)
- System loading requirements, by pin
- Operating clock speed, number of clocks
- I/O definition including pinout and enable for Tri State, and BiDirectional buffers
- Define critical paths
- Define asynchronous behavior

Documentation

- Full hierarchical schematics
- Clocktree and reset diagram
- v3 report
- Test Vector Checker report
- Timing diagrams showing relationship of clocks to data applied and outputs valid

Translation Process

Once the required inputs have been received, the netlist is read into Synopsys. Using Synopsys, each cell of the original design is mapped into an equivalent cell in the Atmel cell library. Atmel defines an equivalent cell as one having not only the same function, but the same speed as the original cell. Sometimes more than one Atmel cell is required. For instance, if the original design was implemented in 1.2µ CMOS, added delays may be required to match the timing of the original cell to avoid race conditions. By matching the timing of each individual cell as closely as possible, time consuming modifications of the design at the netlist level are avoided.

If performance needs to be improved in a particular area, Atmel will address this issue before the Preliminary Design Review. For instance, several cells in the original design may have been used to construct a particular register. To improve performance, Atmel may elect to implement the register as a hard macro. In parallel with the mapping of the netlist, the functional simulation vectors are translated into Atmel format using TDS™ software.

Design Translation Flow

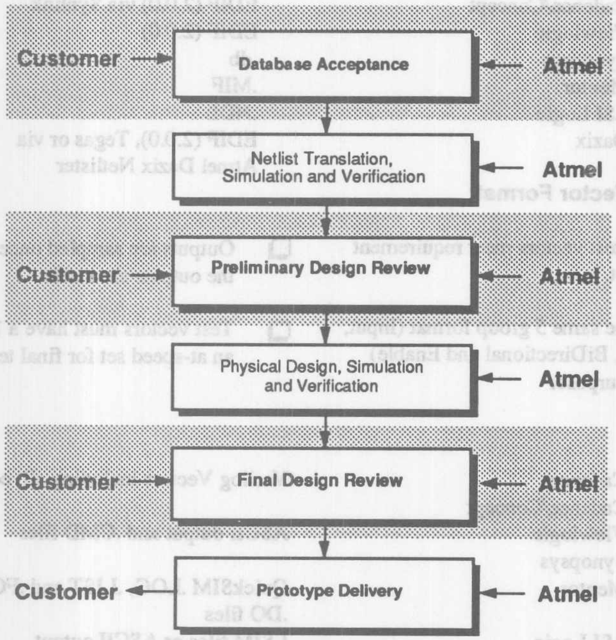


Figure 3-2

Gate Array/Standard Cell Translation

The Atmel ASIC design translation flow is presented in Figure 3-2. In order to execute the design translation, the following design inputs are requested from the customer.

After completing schematic capture and simulation, the customer delivers a design database to Atmel. The following is the data required:

Atmel Design Translation Database Acceptance Check List

Database

Netlist Format

Flatten netlist in Acceptable Format:

Cleanly run v3 Netlist Checker

Cadence	Verilog
Cadence/Concept	EDIF (2.0.0) via Synlink
Viewlogic	EDIF (2.0.0)
Synopsys	.db
Mentor	.MIF
LSI Logic	.NDL
Dazix	EDIF (2.0.0), Tegas or via Atmel Dazix Netlister

Simulation and Test Vector Format

- ❑ All simulation and test vectors meet requirement of Tester Check List.
- ❑ All vectors are in the same 5 group format (Input, Output, Tri State™, BiDirectional and Enable) and have a stated purpose.
- ❑ Outputs are sampled once per clock cycle when the outputs are stable.
- ❑ Test vectors must have a 1 MHz set for probe and an at-speed set for final test.

Acceptance Formats:

Cadence	Verilog Vectors including all control files
Cadence/Concept	
Viewlogic	ASCII output and .CMD files
Synopsys	
Mentor	QuickSIM .LOG, .LIST and .FORCE files including .DO files
LSI Logic	LSIM files or ASCII output
Dazix	ASCII Vectors and time stamp

Specifications

- Operating conditions; Temperature and voltage, best, nominal, worst, (case)
- System loading requirements, by pin
- Operating clock speed, number of clocks
- I/O definition including pinout and enable for Tri State, and BiDirectional buffers
- Define critical paths
- Define asynchronous behavior

Documentation

- A description of the original cell library
- As routed delay from the original design
- Full hierarchical schematics
- Clocktree and reset diagram
- Timing diagrams showing relationship of clocks to data applied and outputs valid

Verifying Performance

The new netlist and translated vectors are used to perform functional simulation. Atmel uses Cadence's Verilog-XL as its "golden simulator." The first step in verification is running the functional simulation vectors. Serial vectors for testability functions and Automatic Test Pattern Generalization for fault grading can also be run at this time. Valid ATPG simulations require that the scan connections be explicitly defined in the netlist. Once functionality has been established, timing verification begins. Atmel will again use Verilog-XL to run the functional vectors at system speed over worst case environmental conditions (voltage and temperature). This can be done for commercial, industrial or military applications. Increasing clock frequency to failure (Fmax), can also provide a measure of design margin. This is performed by running functional vectors in strobe format, rather than print-on-change format.

The timing performance of the original device, either via simulation or Atmel characterization of the sample part, is used for a direct comparison. Atmel performs a waveform comparison of the timing. The system can be programmed to compare timing and report any difference. The net with a greater than desired timing difference can then be sped up or slowed down to meet specification. At this point, Atmel works with the customer to improve performance of a given critical path or to give greater design margin.

Veritime, another Cadence design tool, can be used to verify critical paths (input to register, register to register and register to output) for which timing specifications exist. As a path trace tool, Veritime will identify timing for all paths on the chip, independent of test vectors.

There are other timing-critical simulations that can be run. Pulse width checks on the clocked elements and glitch detection can be performed to ensure proper latch and register transfers after routing. SPICE simulation of the I/O buffers can be run to ensure a match with the switching characteristics. In addition, the clock latency, input set-up times, input hold times and clock-to-output delays can be compared to the original design to ensure there are adequate design margins.

FPGA/PLD Conversion

When to Convert

There are four instances when converting from an FPGA/PLD to a gate array offers the user a direct benefit.

1. High Volume - If your annual volume is over 10,000 units for a single or combined design (commercial), converting to a gate array can save money.

2. Performance - Gate arrays have lower stand-by current requirements and offer greater speed than an FPGA/PLD. Frequently the speed of an FPGA/PLD becomes the limiting factor in a system design.

3. Integration - Converting to a true gate array allows several FPGA/PLDs to be integrated into a single chip, requiring less board space and providing a cost savings.

4. Prototyping - Using the FPGA/PLD for prototyping and then converting into a gate array for production gives the user the best of both worlds; fast design cycle times and low-cost volume production.

The Conversion Process Summary

Atmel's conversion process is designed to minimize the amount of engineering support required from the system designers. Figure 3-3 outlines the conversion process flow. The inputs required vary depending on the original manufacturer of the FPGA/PLD.

design is converted into an equivalent netlist using the Atmel cell library. The equivalent netlist ensures that both the functionality and timing of the new design match the original. Using this technique, almost any FPGA/PLD design can be converted to a gate array.

The original test vectors are also converted and are used to verify the gate array design. Good functional vectors must be provided or developed. This is important because the functional test vectors are the verification vehicle for the design. After the design has been converted and verified for functional performance, the optimization process begins. The design can be optimized to match the timing performance of the original FPGA/PLD design or to meet new performance goals. Additional logic functions or memory can be added to the gate array as well.

Before physical design of the chip begins, a joint Preliminary Design Review is held with Atmel and the customer to approve the results of the converted design. From this point on, the design process is identical to that of a placed and routed on the gate array and verified for electrical and design rules. Atmel uses Cadence's Verilog-XL™ as a golden simulator. Atmel guarantees performance equal to or better than that predicted by Verilog-XL post route simulation.

Back annotation data is extracted from the actual layout and incorporated into the post-route functional and timing simulation. Minor layout modifications may be required to meet the timing specification. A Final Design Review is held to approve the post-route simulation data. After customer approval, the design is released for mask generation and prototyping. Prototypes can be delivered in as little as three weeks and production units in as short as six weeks after customer approval of prototypes. Atmel guarantees the gate array will be a pin-for-pin compatible replacement for the FPGA/PLD.

The exact database requirements for each type of FPGA/PLD are listed on the following pages.

FPGA/PLD Conversion Flow

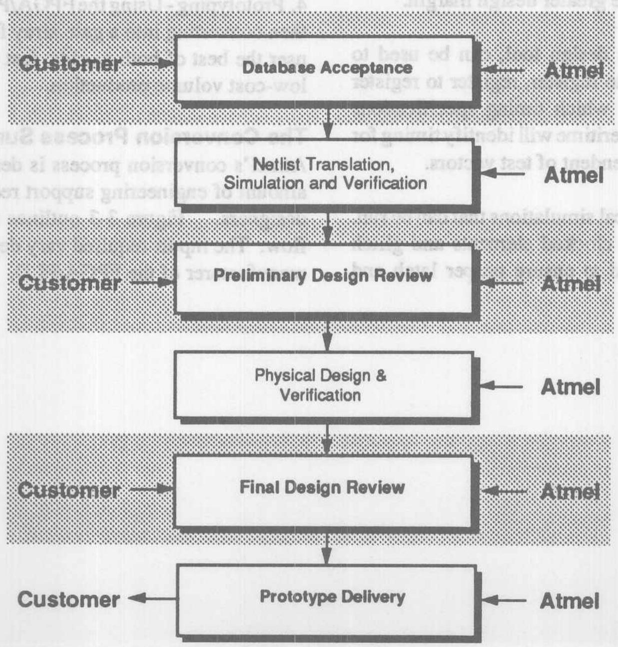


Figure 3-3

Gate Array Database Acceptance Check List

FPGA/PLD To Gate Array Conversion

Database

Converting Actel FPGAs

There are several third party design platforms and tools that support programming of an Actel FPGA, including Mentor, Cadence/Concept, and Viewlogic. The required inputs from each of these platforms are listed below.

Mentor Files
- .MIF (Mentor Interface File - Netlist)
- .LOG (Simulation Log File)
- .LIST (Simulation Listing File)
- .FORCE (Simulation Force File)

Cadence/Concept Files
- EDIF Netlist (via Synlink)
- ASCII File of Simulation Vectors

Viewlogic Files
- .ADL (Flattened Netlist)
- .CMD (Simulator Command File)
- .EDN (EDIF Netlist)
- .PIN (Final Pin List)
- .VSM (Viewlogic Simulator Netlist)
- .SCH (Schematics)
- .SYM (Symbols)
- .WIR (Wire List)

Additional files that can be used to satisfy some of the other data requirements include:

Actel Files
- .CRT (Criticality File)
- .DEL (Actual Delay File)
- .FUS (Fuse File)
- .IPF (Initial Placement File)
- .DEF (Parameters File)

Converting Xilinx FPGAs

There are several third party design platforms and tools that support programming of a Xilinx FPGA, including Mentor, Cadence/Concept, and Viewlogic. The required inputs from each of these platforms are listed below.

Mentor Files
- .MIF (Mentor Interface File - Netlist)
- .LOG (Simulation Log File)
- .LIST (Simulation Listing File)
- .FORCE (Simulation Force File)

Cadence/Concept Files
- EDIF Netlist (via Synlink)
- ASCII File of Simulation Vectors

Viewlogic Files
- .ADL (Flattened Netlist)
- .CMD (Simulator Command File)
- .EDN (EDIF Netlist)
- .PIN (Final Pin List)
- .VSM (Viewlogic Simulator Netlist)
- .SCH (Schematics)
- .SYM (Symbols)
- .WIR (Wire List)

Additional files that can be used to satisfy some of the other data requirements include:

Xilinx Files
- .LCA
- .XNF (Xilinx Netlist)

Gate Array Database Acceptance Check List

FPGA/PLD To Gate Array Conversion

Database

Converting Atmel FPGA/PLD
The specific file requirements for converting from an Atmel PLD are quite straightforward. The ABEL, CUPL or other JEDEC based development system files plus the ASCII file of the vectors define the minimum data required.

JEDEC Files (PLD)
- ABEL
- CUPL
- LOGIC

Viewlogic Files (FPGA, PLD)
- .GDF (Graphic File)
- .TDF (Text File)
- .HIF, .FIT (Fitter File)
- .POF (Programmer Object File)
- .SNF (Simulation Netlist File)
- .VEC (Simulation Vector File

IDS File (FPGA)
- .CDB

Converting Altera FPGA/PLD
Converting from Altera FPGA/PLD into an Atmel gate array follows much the same process as converting from an Atmel FPGA/PLD. The specific file requirements for converting from an Altera FPGA/PLD are the ABEL, CUPL or other JEDEC based development system files plus the ASCII file of the vectors which define the minimum data required. Table 5 lists the recommended Atmel gate array for both the Max Series and the Classic Series of Altera FPGA/PLD. When conversion from a Max series device is desired, there are other files, generated from the MAX+PLUS design system, that can be provided to satisfy some of the data requirements.

Using Altera inputs, an FPGA/EDIF 2.0.0 netlist with actual delay data can be easily extracted. This netlist comes from the MAX+PLUS design system and incorporates the delay data from the fitter/assembler. These files are:

MAX+PLUS Files
Archive File (Menu: File/Project/Archive) should contain these files as a minimum:
- .GDF (Graphic File)
- .TDF (Text File) .HIF, .FIT (Fitter File)
- .POF (Programmer Object File)
- .SNF (Simulator Netlist File)
- .VEC (Simulation Vector File)
- .RPT (Report File)

Design Synthesis

Atmel can accept HDL level inputs in either VHDL or Verilog-HDL languages, incorporating behavioral, data flow and structural level language constructs. A list of VHDL constructs supported by Synopsys v3.0 is given in the VHDL Constructs table.

Atmel uses Synopsys to perform design synthesis, translation from RTL design to a netlist, and design optimization. Atmel will also use Synopsys' Test Compiler to develop a set of functional vectors for simulation and test. If a customer has Synopsys CAE tools, Atmel can provide a timing library and the customer can perform design synthesis. The input to Atmel would then be a netlist and test vectors.

After netlist simulation on Verilog-XL™, the PDR will be held. At PDR the gate-level netlist, simulations and test vectors will be reviewed and agreed to by both parties.

Design Synthesis Flow

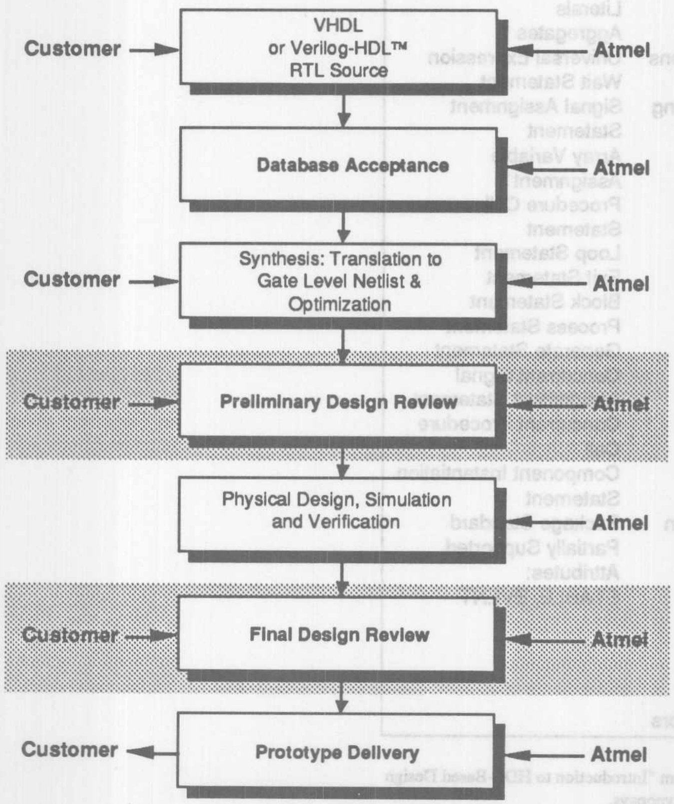


Figure 3-4

4

Fully Supported Constructs:	
Package Declarations	Static Expressions
Package Bodies	If Statement
Constant Declaration	Case Statement
Simple Names	Nest Statement
Operator Symbol	Return Statement
Logical Operators	Null Statement
Relational Operators	Attributes:
Signing Operators	RIGHT, HIGH, LOW,
Qualified Expression	BASE, LEFT, RANGE,
Type Conversion	LENGTH

Synthesis-Constrained Constructs:	
Entity Declarations	Literals
Architecture Bodies	Aggregates
Subprogram Declarations	Universal Expression
Subprogram Bodies	Wait Statement
Subprogram Overloading	Signal Assignment
	Statement
Functional Call	Array Variable
	Assignment
Libraries	Procedure Call
	Statement
Enumeration Types	Loop Statement
Integer Types	Exit Statement
Arrays	Block Statement
Indexed Names	Process Statement
Slice Names	Generate Statement
	Concurrent Signal
Full Type Declaration	Assignment Statement
Subtype Declaration	Concurrent Procedure
Signal Declaration	Call
Variable Declaration	Component Instantiation
Interface Declaration	Statement
Component Declaration	Package Standard
Attribute Declaration	Partially Supported
Attribute Specification	Attributes:
Selected Names	STABLE, EVENT
Attribute Names	
Addition Operators	
Mulitplying Operators	
Operator Overloading	
Miscellaneous Operators	

Gate Array Design Flow

Atmel's design flow has four major milestones independent of the design interface used.

Database Acceptance
Preliminary Design Review
Final Design Review
Prototype Delivery

Atmel has defined specific requirements for each milestone that must be accomplished prior to moving on in the design flow.

Database Acceptance (DA)

At this milestone Atmel formally accepts the design database as complete and begins work on the design. At DA Atmel will verify that the complete database has been received, there are no known errors (netlist checker and test vector checker) and that the netlist, vectors, etc. are all in the correct format.

It is critical that all the required data elements are submitted at one time in the correct format and that the database is final (no anticipated changes).

- Database Summary
- Flat netlist (at level netlist if test compiler will be used)
- TV's per Atmel's test vector checklist.

Documentation
- v3 Netlist Checker Report
- Test Vector Checker (tvc) Report
- Timing Diagram (clocks, vectors applied, vectors sampled)
- Critical Paths
- Best/Worst Case Conditions
- System Loading Requirements
- Purpose of TV Sets

A detailed database acceptance checklist, for each of the four design flows (cell library, ASIC translation, FPGA/EPLD conversion and VHDL/Verilog-HDL) provides the designer with the exact requirements.

Preliminary Design Review (PDR)

After DA, Atmel will migrate all designs into the Cadence Design System. Atmel uses Cadence's Verilog-XL/Veritime as our golden simulator and all designs are signed off based on the Cadence results. The submitted design is ported to Verilog and re-simulated using the customer supplied vectors. Functional performance is verified as well as key timing performance. Once this has been successfully completed, a PDR is held to agree upon the performance level of the gate array. Following are the requirements for PDR:

Confirm v3 and tvc files correct
I/O buffer listing and bonding diagram
Preliminary testability compiler report
Route clock tree and analysis of worst case and best case delay
Verilog simulation at-speed
 -nominal, worst case, best case (with no timing violations)
Review critical path information (t_{SU}, t_{HOLD}, t_{PD})
 -Verilog or Veritime estimates
I/O electrical specifications
Electromigration calculation

Final Design Review (FDR)

The Final Design Review is the last joint review between Atmel and the customer before committing to prototypes. FDR is basically a re-review of PDR. Prior to the meeting both Atmel and the customer will have reviewed the post-route Verilog-XL simulation incorporating the back annotation data. The customer will receive back annotation data for complete post-route simulation on their CAE systems. Atmel guarantees silicon performance equal to or better than predicted on the post-route Verilog-XL simulations. Below are the requirements for FDR.

Updates of cell mapping and timing (if any)

Post-route netlist check v3, tvc
 -post-route netlist changes
Post-route timing simulation to specification
 -review clock timing
 -at speed
 -clock skew (if required)
 -listing of timing warnings with explanation

Static path analysis (as specified)
Electromigration Calculation
Bonding diagrams and pin list
 -bond pad plot

LVS/DRC/ERC

Prototype Delivery

Atmel will deliver 10 prototypes in ceramic packages to the customer. The purpose of the units are to verify the design's functionality and electrical performance.

PDR Checklist

- Confirm v3 file correct
- O.L. Nets, Open Nets, Multiple Drivers
- I/O buffer listing and bonding diagram
- Preliminary testability compiler report indicates areas we expect are not covered
- Route clock tree and analysis of worst case and best case delay
- Verilog simulation - 1 speed
  Nominal, worse case, best case
  (with no hold time violations)
  - review clock timing
  - at-speed
  - clock skew (if required)
  - listing of timing warnings w/explanation
- Review critical path information
  (t_{SU}, t_{HOLD}, t_{PD})
  - Verilog or Veritime estimates
- I/O Electrical
- Electromigration calculation pre-FDR
- ASIC Translation

  Minicell List
  - cells used
  - Atmel mappings
  - notation for new cells and new mappings

  I/O buffer list with same information as minicell list
  - I/O buffer Atmel comparisons

- Timing comparison for all cells
- New soft macro mappings
- Document netlist changes due to mapping

FDR Checklist

- Confirm v3 file correct
  - Post-route netlist changes
- O.L. Nets, Open Nets, Multiple Drivers
- I/O buffer listing and bonding diagram
- Testability compiler report
- Clock tree and analysis of worst case and best case delay
- Verilog simulation - with back annotation
  Nominal, worse case, best case
  (with no hold time violations)
  - review clock timing
  - at-speed
  - clock skew (if required)
  - listing of timing warnings w/explanation
- Review critical path information
  (t_{SU}, t_{HOLD}, t_{PD})
  - Verilog or Veritime results
- I/O Electrical
- Electromigration calculation
- Ground bounce
- LVS, DRC

Overview of Atmel Gate Array v3 Netlist Checker

☐ No Q → D Connects

Use the 'd' command in the net check menu. Register outputs directly connected to the inputs of other registers should be buffered to slow down the path, so that the data will not precede the clock and be latched one clock early. If the registers are on different clocks, simulations should verify that data is latched at the correct time. If the registers are on the same clock, the simulation will apply the clock to the register clock pins simultaneously when in actuality there will be some lag time. This condition cannot be verified in simulation and must be either waived by the customer or corrected.

☐ No Fanout Ratio Violations

Use the 'f 1.5' for military conditions, 'f 2.0' for commercial in the net check menu. Fanout violations which occur on specially routed clock tree may be waived if the calculated loading delay plus time-of-flight numbers yield an acceptable clock latency and skew. Other fanout ratios greater than the limits should be corrected by adding either a high drive version of the driving cell or a buffer, knowing that timing will be slower as a result. Atmel's post-route fanout ratio is 2.5 for both military and commercial customers.

A new method for checking fanout based on slope of the driving cell will be available shortly in new releases of v3.

☐ No Floating Inputs

This applies to internal cells in the gate array. Floating CMOS inputs spell trouble and MUST be tied high or low before going to route. Upon entering the check menu, the net summary will report on the number of floating inputs found in the design. Use the '1' command for a detailed report on which pins are floating.

☐ Tristate Busses Have HLDs on Them

As a special subset of the previous check for floating inputs, all internal tristate busses should include a weak bus hold cell, the HLD1, which sustains the last applied logic value if all tristate cells enter tristate mode. If internal tristate busses are present in the design, use the 't' command in the net check menu to list the name of each net, then the 'n <num>' command to check for the presence of an HLD1 on that net.

☐ No "confused" Nets

A confused net either has

1) multiple unmatched drivers (non-parallel)
2) mixed regular, tristate, and/or bidirectional drivers
3) pins with unknown attributes (from unknown cells)
4) incorrectly defined I/O (at the top level)
5) any other illegal combination of pins tied together

Confused nets are listed whenever the user enters the check menu. The multiple drivers report 'm/e' can help to identify some problems, and entering the check menu using the 'ce' command from the top menu can also be of some help. Confused nets sometimes originate from an improperly generated netlist. Many times a top level symbol is needed if the netlist is being generated from a schematic capture tool.

☐ Chip Fits on Target Array

The 'me' command in the top menu will size the design against each standard gate array. Although the cutoff percentage is 90% for each array, occasionally arrays have been utilized up to 95%. The I/O count will not be accurate since it does not take power and ground pads into account. This should be checked separately given a pad placement file. The 'me' command is automatically run every time interactive mode is entered.

☐ Cells Have Layout

The v3 libraries have been checked against the router databases, and only cells which have layout and timing are entered. If the netlist reader portion of v3 does not give any errors due to unknown cells, then all cells in the design have layout.

☐ No Mixed PAD Libraries

Only one pad library at a time may be loaded into v3. Currently Atmel's pads come in three distinct flavors:

1) the ATL PD and PS regular pads
2) the ATL "C" PADC commercial pads
3) the ATL80 submicron pads

Pads from these separate libraries may not be used together in the same design. The netlist reader section of v3 will only recognize one type of pad at a time.

Electromigration Checked

Use the 'e [freq] <units>' command in the net check menu. The check is meant to be a worst case prediction in which the logic state of the net changes twice every clock period. This happens with clock signals, but most other signals will only change once per clock period in a synchronous design. If the net changes states less frequently, the electromigration current (in mA) may be multiplied by the frequency of change to lower the final number. Nets which exceed the current limits should be flagged for the route, so that the line widths can be increased.

No Net Aliasing Problems

Use 's' in the back annotation menu to check for net aliasing. Net aliasing can occur when writing the netlist out for routing, because illegal characters are filtered out or converted to other legal characters. If an error message comes up, then the netlist must be written in indexed notation to DEF. This will cause problems later on if an ECO/ECN changes the number or order of the nets in the design. Use this option with caution. To read the back annotation information from the router back into v3, use the 'li <file>' option in the back annotation menu (load indexed router report).

A second case of net aliasing occurs in the netlist itself, not in the interface between v3 and the router. Use the 'a' command in the check menu to report on power, ground, and other aliased nets. There should be no other aliased nets, or problems will occur.

No Instance Aliasing Problems

Use 's' in the design prober menu to check for instance aliasing. In a similiar fashion to net aliasing, illegal characters in the netlist may be filtered out when targeting the netlist for the router. Make sure there are no instance aliasing problems. If there are, instance names should be edited to guarantee uniqueness.

All Parallel Cells are Intentional

Parallel cells occur most frequently in clock distribution networks, usually to obtain adequate drive. Check for parallel and mixed drivers with the 'm' command in the check menu. The 'me' command provides more details. v3 can detect many but not all instances of multi-stage parallelism, where the logic remains parallel but does not recombine during the intermediate stages, coming together only at the end.

Gate Array Test Application Check List

In order to tranfer new designs from design to prototype and production as quickly as possible, customers should adhere to the following when preparing vectors for test applications.

Simulation:

- ☐ All vector files must be in Tabular ASCII format . This allows the use of proprietary software and thus reduces the amount of hand editing and conditioning.
- ☐ Although the preferred format is the ASCII file mentioned above, the following Simulators are presently supported directly through TDS. This capability is available if the files cannot be supplied in ASCII format. (CADAT, QUICKSIM, HILO, LASAR, LSIM SILOS, TEGAS, TEXSIM, VALIDSIM, VERILOG, ZILOS)
- ☐ Simulation log files can be supplied on 8 mm, 1/4 cartridge or 9-track tapes. Tapes should be in tar format. If compression is used it must be noted in the documentation.
- ☐ Time based simulation including all external signals and internal nodes or enables for bi-directional and tri-state buffers.
- ☐ Pull-ups must not be used in the simulation. It must be clear in the logfile when a signal on the device is tristated.
- ☐ Best and worst case simulations will insure that the output data does not cross cycle boundaries.
- ☐ Simulation to include temperature and voltage margins.

Timing:

- ☐ Timing diagrams, representing the input and output timings used in the simulation, are essential.
- ☐ The hardware specifications listed in the ASIC Test Capability Reference Table in the design manual should be followed for all timing.
- ☐ The simulation must be run at the intended operating frequency. If the timing cannot be scaled down for wafer testing, an additional simulation must be run at 1 MHz.
- ☐ Maximum Frequency must not exceed 50 MHz (256 I/O Channels) or 100 MHz(128 I/O Channels). It is preferred to test at speeds below 40 MHz and guarantee speed through delay parameters.
- ☐ Simulations must be performed such that the pattern can be broken down into"cycles" during post processing. For optimum performance the test system should apply test vectors on a fixed periodic basis.
- ☐ One time set should be used if possible. This greatly reduces debug/parametric measurement efforts.
- ☐ The timing should be setup with the expected data sampled before the systemclock and the input data applied after system clock. This gives the mosttime for setup and propagation.
- ☐ All expected output files must contain information starting at time 0 ps. The conversion software used requires a 0 reference point for subsequent vectors.
- ☐ Minimum signal width must not exceed 10ns.
- ☐ The pattern must not contain multiple clocks per cycle.
- ☐ The pattern must not contain changes of format on-the-fly.

4

Signal Specifications:

☐ A complete cross-reference list of all signals used in the simulation. This should allow renaming the signals to match the test specification and should also state which signals control other signals.

SIGNAL NAME Simulation	Specification	I/O/B	Buffer	Enable
sample	SAMPLE	B	PDB4U	/$internal1

☐ Pull-ups must not be used in the simulation. It must be clear in the logfile when a signal on the device is tristated.

☐ A single signal must be made available to determine the direction of bi-directional buffers.

☐ A single signal must be made available to determine the state of tri-statable buffers.

☐ Outputs must not be tested until the device has been fully initialized, even if some outputs are in known states.

Test System Considerations:

☐ At the beginning of each simulation file there is to be a set of vectors that will reset the part so it will be in a known state. Each simulation file must be a "stand alone" module. States must be achieved by using the test pattern to initialize the device. Power-On-Reset circuitry must not be used for this purpose. The reset section is to be identified as to exactly where it ends and the actual vectors begin.

☐ Direction changes on bi-directional signals must occur on a cycle boundary on the test system. The simulations should be run in such a way that this can be accomplished. (see Figure 3).

☐ Data must be strobed at the same point in each cycle with only one strobe point per cycle.

☐ A maximum of 6 input clocks can be used.

☐ All bi-directional pins are to be in a tri-state state for at least one cycle when switching between input/output or output/input. This will help eliminate bus contention during test.

☐ Quantity of cycles not to exceed 32,000 per module.

Parametric Testing:

Vectors should be supplied that setup the chip for all requiredparametric measurements. These tests can be combined into one pattern if the locations at which to make the measurement are explicitly stated.

☐ Test patterns are needed specifically for testing DC parameters.

VOL/VOH- All outputs should achieve both a low and a high state for a minimum of 2 cycles. This will allow stopping on the appropriate test vector to make the measurement.

VIL/VIH- All inputs should be exercised low and high in such a way that the output of the chip ischanged. In order to get a reading for VIH or VIL, that signal must cause a change on another signal that is being compared in the pattern.

IDD- Static and Dynamic, condition the device as required. This pattern should set all the inputs to a know state and disable bi-directional/tristate buffers.

Tristate- Functional or Parametric, all bi-directional and tri-state outputs should achieve a high impedance state from both a low and high input, and or output state.

☐ No AC tests will be performed unless prior negotiation is done.

☐ Temperature testing to be done within standard military or commercial ranges.

Note: Final acceptance of the simulation waveforms will be determined upon analysis with Atmel's Test VectorChecker (tvc). The tvc analysis will determine if the timing used meets the constraints of the test system and will also indicate if all parametrics are covered.

Overview of Atmel Test Vector Checker

The Atmel Test Vector Checker (tvc) is proprietary software used to check the simulation test vectors for compatibility with present test systems. The software analyzes the timing and states of all signals and reports on any items that may not be readily duplicated on the test system. The software also checks the simulation for parametric coverage and general details of tester utilization. The table below lists the individual checks.

Timing Checks

- ❑ Delay used with non-return-to-zero not consistent
  This warning indicates that the delay for a NRZ waveform changed during the simulation for a given signal.
- ❑ Delay used with return-to-zero not consistent
  This warning indicates that the delay for a RZ waveform changed during the simulation for a given signal.
- ❑ Width used with return-to-zero not consistent
  This warning indicates that the width for a RZ waveform changed during the simulation for a given signal.
- ❑ Delay used with return-to-one not consistent
  This warning indicates that the delay for a RTO waveform changed during the simulation for a given signal
- ❑ Width used with return-to-one not consistent
  This warning indicates that the width for a RTO waveform changed during the simulation for a given signal.
- ❑ Both NRZ and RZ waveforms used for the same signal. This warning indicates that a RZ formatted signal was allowed to stay HIGH for a cycle.
- ❑ Both NRZ and RTO waveforms used for the same signal. This warning indicates that a RTO formatted signal was allowed to stay LOW for a cycle.
- ❑ Inputs floating during cycle
  This warning indicates that a given input is not forced to a valid state within a cycle.
- ❑ More than one I/O change within a cycle
  This warning indicates that the signal changed direction more than one time within a given cycle.

Parametric Coverage Checks

- ❑ Checks for all valid states on all signals. The report will show only those states that are not achieved in the simulation.
- ❑ Checks for all valid transitions on all signals. The report will show only those transitions that do not occur in the simulation.
- ❑ Indicate if bidirectional buffers never change direction.

Tester Utilization

- ❑ Total pattern depth (without compression).
- ❑ Total number of input, output and bidirectional signals.
- ❑ Total number of tester timing generators needed.
- ❑ Histogram of output transitions within cycle to indicate proper strobe position.

Design for the Synopsys Test Compiler

The following guidelines are generated to allow designs to be efficiently enhanced by using the Synopsys Test Compiler software. If all design rules are followed, the Test Compiler can add scan chains to the design and generate test vectors automatically for nearly 100% fault coverage. When a design rule is not followed, fault coverage is reduced, sometimes dramatically, unless the design is modified to comply with the rules. This approach will require at least 1 additional pin for Test Enable (TE) and at least 2 pins if muxing is required for a Test Enable (TE) and Test (T) mode.

Synopsys Test Compiler Guidelines

Testability Rule	Effects of Infraction	Workaround
Synchronous Design -No cross coupled gates -No unregistered feedback	Associated logic untestable	Break feedback path with test mode
Single Edge Clocking	Clocked device not allowed in scan chain - reduced fault coverage	In test mode, create single edge clocking with inverters and MUXs
No Clock Gating	Clocked device not allowed in scan chain - reduced fault coverage	Use data disable flip-flops instead of clock enables, disable gating in test mode
No Latches	Not allowed in scan chain, reduced fault coverage	Use alternate test methods, force latches to transparent mode with test mode
Single External Reset -No asynchronous resets or presets generated on chip -No combinational logic in reset path	Not allowed in scan chain, reduced fault coverage	Reset OR'd with test mode
No Internal Tristate Buses	Reduced fault coverage, possible tristate contention during scan test	Use MUXs or AOI gates, insert gating of controls to prevent contention
No Direct Q to D Connections	Dynamic Hazard	

How to Use This Cell Library Index

The cell index contains the macro cell's timing, size, and loading information. The data included in the cell timing information is explained in detail below. A separate cell index is provided for each technology (ATL-1.0 µ and ATL80-0.8 µ).

Cell Parameters

Site Count: Lists the number of gate array cell sites the macrocell occupies. This can be used by the designer to determine what size gate array is required for the design.

Input Load: Lists the number of unit loads the input represents. The number of unit loads is used to determine the propagation delay of the macrocell. This is covered in the timing section.

For initial logic design, a load of less than 12 is recommended. The logic load should be supplemented with estimated wire load (pre-route) or actual back annotated wire load (post-route). With estimated wire loads included, a load of less than 24 is recommended. After routing, a back annotated load of 30 is considered a limit. Loads greater than this will be addressed by inserting higher drive cells, reducing wire load or waiving the violation for a non-critical net (such as an external reset that is neither timing critical nor functionally subject to improper operation due to noise).

Timing

Path Timechk: Shows the path for the associated timing numbers. Path Timechk "A->O" denotes the delay from input A to output O. Path Timechk "D+CLK" denotes the Setup/Hold requirement between inputs D and CLK.

The cell index gives the designer the information needed to calculate the propagation delay based on the number of loads the macrocell is driving. The delay is calculated using the Delay = Intercept + (Slope x Loads) equation where:

- Delay is the total propagation delay.
- Intercept is the base (zero loads) delay.
- Slope is the additional delay per unit load.
- Loads is the number of unit loads (not macrocells) the macrocell is driving.

The number of unit loads each macrocell's input represents is listed under Input Load in the cell index.

74XX Series Soft Macros

Cell Name	Cell Description MSI Part Number	Similar to MSI Part Number
TTL64	4232 And Or Invert	74ALS64
TTL74	DFF	74ALS74
TTL109	JKBar FF	74ALS109
TTL138	3 to 8 Decoder	74ALS138
TTL139	Dual 2 to 4 Decoder	74ALS139
TTL148	8 to 3 Priority Encoder	74ALS148
TTL151	8 to 1 MUX	74ALS151
TTL153	4 to 1 MUX	74ALS153
TTL157	Quad 2 to 1 MUX	74ALS157
TTL158	Quad Inverting 2 to 1 MUX	74ALS158
TTL161	4 BIT Binary Counter	74ALS161
TTL164	8 BIT Serial in Parallel Out Shift Register	74ALS164
TTL166	8 BIT Parallel Load Shift Register	74ALS166
TTL169	4 BIT U/D Counter	74ALS169
TTL174	Hex DFF	74ALS174
TTL194	4 BIT BiDi Shift Register	74ALS194
TTL273	Octal DFF	74ALS273
TTL280	9 BIT Parity Generator	74ALS280
TTL283	4 BIT Fast Adder	74ALS283
TTL688	8 BIT Comparator	74ALS688
TTL823	9 BIT DFF	74ALS823
TTL825	8 BIT DFF	74ALS825
TTL85	4 BIT Magnitude Comparator	74ALS85
TTL86	Quad 2 Input Ex OR	74ALS86

ATL 1.0 µ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Macrocells in alpha-numeric order

				Intercept		Slope	
Signal Name	Description (Site Count)	Path Timechk	Input Load	Rising Hold-R	Falling Hold-F	Rising Setup-R	Falling Setup-F
ADD3	1 bit full adder (6)	P->CO	1	1.681	1.782	0.121	0.110
		P->SO	1	0.878	1.075	0.099	0.062
		Q->CO	4	0.497	0.576	0.103	0.073
		Q->SO	4	0.652	0.725	0.090	0.062
		CI->CO	3	0.728	0.785	0.091	0.069
		CI->SO	3	0.400	0.596	0.089	0.061
ADD3X	1 bit full adder with buffered outputs (4)	P->CO	2	1.245	1.387	0.060	0.048
		P->SO	2	1.175	1.078	0.060	0.050
		Q->CO	2	0.605	1.164	0.061	0.047
		Q->SO	2	1.243	0.977	0.062	0.048
		CI->CO	2	0.730	0.714	0.055	0.050
		CI->SO	2	0.772	0.581	0.059	0.047
AND2	2 input AND (1)	A->O	1	0.439	0.501	0.056	0.041
		B->O	1	0.485	0.423	0.056	0.043
AND2H	2 input AND - high drive (1)	A->O	1	0.554	0.564	0.028	0.024
		B->O	1	0.554	0.564	0.028	0.024
AND3	3 input AND (2)	A->O	1	0.689	0.572	0.058	0.046
		B->O	1	0.689	0.572	0.058	0.046
		C->O	1	0.689	0.572	0.058	0.046
AND3H	3 input AND - high drive (2)	A->O	1	0.732	0.618	0.032	0.025
		B->O	1	0.732	0.618	0.032	0.025
		C->O	1	0.732	0.618	0.032	0.025
AND4	4 input AND (2)	A->O	1	0.806	0.642	0.063	0.043
		B->O	1	0.806	0.642	0.063	0.043
		C->O	1	0.806	0.642	0.063	0.043
		D->O	1	0.806	0.642	0.063	0.043
AND4H	4 input AND - high drive (2)	A->O	1	0.861	0.647	0.038	0.026
		B->O	1	0.861	0.647	0.038	0.026
		C->O	1	0.861	0.647	0.038	0.026
		D->O	1	0.861	0.647	0.038	0.026

Signal Name	Description (Site Count)	Path Timechk	Input Load	Rising Hold-R	Falling Hold-F	Rising Setup-R	Falling Setup-F
AND5	5 input AND (2)	A->O	1	0.440	0.447	0.138	0.046
		B->O	1	0.440	0.447	0.138	0.046
		C->O	1	0.440	0.447	0.138	0.046
		D->O	1	0.440	0.447	0.138	0.046
		E->O	1	0.440	0.447	0.138	0.046
AOI22	2 input AND into 2 input NOR (1)	A->O	1	0.409	0.348	0.105	0.043
		B->O	1	0.363	0.321	0.101	0.062
		C->O	1	0.435	0.304	0.103	0.063
AOI22H	2 input AND into 2 input NOR - high drive (2)	A->O	2	0.405	0.234	0.046	0.033
		B->O	2	0.323	0.257	0.048	0.035
		C->O	2	0.371	0.275	0.048	0.026
AOI222	Two, 2 input ANDs into 2 input NOR (2)	A->O	1	0.522	0.517	0.104	0.063
		B->O	1	0.608	0.499	0.105	0.065
		C->O	1	0.498	0.327	0.103	0.061
		D->O	1	0.433	0.353	0.103	0.062
AOI222H	Two, 2 input ANDs into 2 input NOR - high drive (2)	A->O	2	0.406	0.428	0.038	0.032
		B->O	2	0.406	0.428	0.038	0.032
		C->O	2	0.406	0.428	0.038	0.032
		D->O	2	0.406	0.428	0.038	0.032
AOI2223	Three, 2 input ANDs into 3 input NOR (2)	A->O	1	0.647	0.643	0.105	0.064
		B->O	1	0.647	0.643	0.105	0.064
		C->O	1	0.647	0.643	0.105	0.064
		D->O	1	0.647	0.643	0.105	0.064
		E->O	1	0.647	0.643	0.105	0.064
		F->O	1	0.647	0.643	0.105	0.064
AOI2223H	Three, 2 input ANDs into 3 input NOR - high drive (4)	A->O	1	0.892	0.780	0.061	0.044
		B->O	1	0.892	0.780	0.061	0.044
		C->O	1	0.892	0.780	0.061	0.044
		D->O	1	0.892	0.780	0.061	0.044
		E->O	1	0.892	0.780	0.061	0.044
		F->O	1	0.892	0.780	0.061	0.044
BUF1	1x buffer (1)	I->O	1	0.372	0.450	0.054	0.040

ATL 1.0 μ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
BUF2	2x buffer (1)	I->O	1	0.461	0.512	0.026	0.021
BUF2T	2x Tri-state™ bus driver with active high enable (2)	I->O	1	0.505	0.677	0.088	0.048
		E->O	1.5	0.282	0.000	0.089	0.065
BUF2Z	2x Tri-state bus driver with active low enable (2)	I->O	1	0.503	0.677	0.088	0.049
		E->O	1.5	0.168	0.402	0.099	0.049
BUF3	3x buffer (2)	I->O	1	0.495	0.547	0.020	0.019
BUF4	4x buffer (2)	I->O	1	0.549	0.624	0.016	0.014
BUF8	8x buffer (3)	I->O	2	0.557	0.636	0.007	0.006
BUF12	12x buffer (4)	I->O	3	0.565	0.643	0.005	0.004
BUF16	16x buffer (5)	I->O	4	0.545	0.618	0.004	0.003
CLA7X	7 input carry lookahead (2)	A->O	1	0.816	0.472	0.230	0.105
		B->O	1	0.813	0.531	0.157	0.092
		C->O	1	0.769	0.354	0.136	0.076
		D->O	1	0.582	0.397	0.119	0.070
		E->O	1	0.657	0.245	0.117	0.063
		F->O	1	0.402	0.317	0.087	0.061
		G->O	1	0.529	0.104	0.089	0.062
DEC4	2:4 decoder (4)	S0->D0	3	0.484	0.580	0.059	0.060
		S0->D1	3	0.263	0.327	0.073	0.072
		S0->D2	3	0.484	0.580	0.059	0.060
		S0->D3	3	0.262	0.327	0.073	0.072
		S1->D0	3	0.524	0.587	0.059	0.059
		S1->D1	3	0.527	0.578	0.059	0.060
		S1->D2	3	0.336	0.237	0.068	0.068
		S1->D3	3	0.329	0.242	0.070	0.069

4

ATL 1.0 μ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
DEC4N	2:4 decoder with active low enable (4)	S0->D0	1	0.473	0.820	0.060	0.062
		S0->D1	1	1.017	0.876	0.060	0.061
		S0->D2	1	0.477	0.882	0.060	0.065
		S0->D3	1	1.042	0.859	0.059	0.061
		S1->D0	1	0.505	0.627	0.060	0.062
		S1->D1	1	0.513	0.605	0.061	0.060
		S1->D2	1	0.784	0.742	0.059	0.061
		S1->D3	1	0.774	0.723	0.060	0.062
		E->D0	2	0.578	0.696	0.059	0.061
		E->D1	2	0.501	0.668	0.059	0.064
		E->D2	2	0.573	0.738	0.060	0.065
		E->D3	2	0.510	0.634	0.058	0.060
DEC8N	3:8 decoder with active low enable (12)	S0->D0	1	0.742	1.154	0.061	0.080
		S0->D1	1	1.525	1.381	0.061	0.080
		S0->D2	1	0.742	1.154	0.061	0.080
		S0->D3	1	1.526	1.381	0.061	0.080
		S0->D4	1	0.762	1.153	0.060	0.078
		S0->D5	1	1.525	1.385	0.061	0.079
		S0->D6	1	0.745	1.155	0.061	0.080
		S0->D7	1	1.526	1.385	0.061	0.079
		S1->D0	1	0.664	0.847	0.062	0.079
		S1->D1	1	0.664	0.847	0.062	0.079
		S1->D2	1	1.067	1.092	0.059	0.078
		S1->D3	1	1.067	1.092	0.059	0.078
		S1->D4	1	0.664	0.840	0.062	0.080
		S1->D5	1	0.664	0.840	0.062	0.080
		S1->D6	1	1.067	1.089	0.059	0.080
		S1->D7	1	1.067	1.091	0.059	0.079
		S2->D0	1	0.733	0.826	0.061	0.081
		S2->D1	1	0.733	0.826	0.061	0.081
		S2->D2	1	0.733	0.827	0.061	0.080
		S2->D3	1	0.733	0.829	0.061	0.080
		S2->D4	1	1.113	1.135	0.063	0.082
		S2->D5	1	1.118	1.135	0.063	0.082
		S2->D6	1	1.117	1.135	0.060	0.081
		S2->D7	1	1.116	1.135	0.062	0.081
		E->D0	2	0.767	1.194	0.058	0.086
		E->D1	2	0.767	1.194	0.058	0.086
		E->D2	2	0.767	1.194	0.058	0.086
		E->D3	2	0.767	1.194	0.058	0.086
		E->D4	2	0.767	1.194	0.058	0.086
		E->D5	2	0.767	1.194	0.058	0.086
		E->D6	2	0.767	1.194	0.058	0.086
		E->D7	2	0.767	1.194	0.058	0.086
DFF	D flip-flop (5)	CLK->Q	1	1.009	0.934	0.058	0.048
		D+CLK	2	0.290	0.000	0.271	0.210

ATL 1.0 µ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
DFFBCPX	D flip-flop with asynchronous	C->Q	2	-----	0.553	-----	0.041
	clear and preset with	C->QB	2	1.156	-----	0.175	-----
	complementary outputs (6)	CLK->Q	1	1.188	0.950	0.106	0.048
		CLK->QB	1	1.218	1.40 5	0.179	0.194
		P->Q	2	1.151	-----	0.175	-----
		P->QB	2	-----	0.687	-----	0.043
		D+CLK	2	0.504	0.035	0.360	0.328
DFFBSRX	D flip-flop with asynchronous	CLK->Q	1	1.028	1.123	0.059	0.073
	set and reset with	CLK->QB	1	1.350	1.309	0.179	0.153
	complementary outputs (6)	R->Q	2	-----	1.092	-----	0.154
		R->QB	2	0.623	-----	0.057	-----
		S->Q	2	0.480	-----	0.056	-----
		S->QB	2	-----	1.008	-----	0.152
		D+CLK	2	0.224	0.500	0.430	0.405
DFFC	D flip-flop with asynchronous	CLK->Q	1	1.069	0.916	0.063	0.049
	clear (5)	C->Q	2	-----	0.896	-----	0.048
		D+CLK	2	0.519	0.124	0.426	0.295
DFFH	D flip-flop - high drive (5)	CLK->Q	1	1.059	0.946	0.029	0.027
		D+CLK	2	0.385	0.069	0.430	0.295
DFFR	D flip-flop with asynchronous	CLK->Q	1	1.079	0.924	0.061	0.049
	reset (5)	R->Q	1	-----	1.215	-----	0.050
		D+CLK	2	0.519	0.170	0.485	0.295
DFFS	D flip-flop with asynchronous	CLK->Q	1	1.006	0.968	0.058	0.057
	set (5)	S->Q	2	0.754	-----	0.058	-----
		D+CLK	2	0.630	0.030	0.343	0.490
DFFSR	D flip-flop with asynchronous	CLK->Q	1	1.113	1.009	0.104	0.055
	set and reset (5)	S->Q	2	0.855	-----	0.104	-----
		R->Q	2	-----	0.346	-----	0.043
		D+CLK	2	0.604	0.055	0.343	0.490
DLY2000	Delay buffer 2.0 ns (3)	I->O	1	2.315	2.166	0.058	0.067
DLY3500	Delay buffer 3.5 ns (5)	I->O	1	3.388	3.542	0.052	0.059

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
DLY8000	Delay Buffer 8.0 ns (11)	I->O	1	8.011	7.637	0.052	0.060
DSS	Set scan D flip-flop (6)	CLK->Q	1	0.826	0.9 20	0.069	0.045
		D+CLK	2	0.000	0.000	0.650	0.450
		TI+CLK	2	0.000	0.000	0.700	0.500
		TE+CLK	2	0.000	0.000	0.930	0.860
DSSBCPY	Set scan flip-flop with clear and preset (11)	CLK->Q	1	0.725	1.013	0.069	0.051
		CLK->QB	1	1.053	1.253	0.050	0.066
		C->Q	2	-----	1.222	-----	0.049
		C->QB	2	0.773	-----	0.070	-----
		P->Q	2	0.709	-----	0.068	-----
		P->QB	2	-----	1.304	-----	0.055
		D+CLK	2	0.000	0.000	0.800	1.000
		TI+CLK	2	0.000	0.000	1.100	1.200
DSSBR	Set scan flip-flop with reset (9)	CLK->Q	1	1.442	1.430	0.068	0.045
		CLK->QB	1	1.150	1.173	0.075	0.060
		R->Q	2	-----	0.713	-----	0.044
		R->QB	2	0.444	-----	0.070	-----
		D+CLK	2	0.000	0.200	0.800	0.800
		TI+CLK	2	0.000	0.100	1.100	1.100
DSSBS	Set scan flip-flop with set (10)	CLK->Q	1	1.563	1.696	0.030	0.022
		CLK->QB	1	1.276	1.165	0.041	0.032
		S->Q	1	1.680	-----	0.028	-----
		S->QB	1	-----	1.281	-----	0.029
		D+CLK	2	0.000	0.000	0.700	1.000
		TI+CLK	2	0.000	0.000	0.900	1.200
DSSR	Set scan D flip-flop with reset (7)	CLK->Q	1	0.929	1.012	0.073	0.046
		R->Q	2	-----	0.639	-----	0.046
		D+CLK	2	0.000	0.000	0.650	0.350
		TI+CLK	2	0.000	0.000	0.650	0.350
		TE+CLK	2	0.000	0.000	1.090	0.850
DSSS	Set scan D flip-flop with set (7)	CLK->Q	1	0.901	1.067	0.070	0.047
		S->Q	1	0.931	-----	0.068	-----
		D+CLK	2	0.000	0.000	0.630	0.650
		TI+CLK	2	0.000	0.000	0.700	0.700
		TE+CLK	2	0.000	0.000	0.930	1.000

ATL 1.0 µ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
DSSSR	Set scan flip-flop with set and reset (8)	CLK->Q	1	1.211	1.274	0.071	0.044
		R->Q	2	-----	0.615	-----	0.044
		S->Q	2	0.948	-----	0.067	-----
		D+CLK	2	0.000	0.000	0.790	0.440
		TI+CLK	2	0.000	0.000	0.800	0.430
		TE+CLK	2	0.000	0.000	1.100	0.750
INV1	1x inverter (1)	I->O	1	0.250	0.193	0.055	0.045
INV1D	Dual 1x inverters (1)	I0->O0	1	0.250	0.193	0.055	0.045
		I1->O1	1	0.250	0.193	0.055	0.045
INV1Q	Quad 1x inverters (2)	I0->O0	1	0.250	0.193	0.055	0.045
		I1->O1	1	0.250	0.193	0.055	0.045
		I2->O2	1	0.250	0.193	0.055	0.045
		I3->O3	1	0.250	0.193	0.055	0.045
INV1TQ	Quad Tri-state inverter (5)	E0->O0	1	0.450	0.558	0.107	0.064
		E0->O1	1	0.450	0.558	0.107	0.064
		E0->O2	1	0.450	0.558	0.107	0.064
		E0->O3	1	0.450	0.558	0.107	0.064
		E1->O0	1	0.450	0.558	0.107	0.064
		E1->O1	1	0.450	0.558	0.107	0.064
		E1->O2	1	0.450	0.558	0.107	0.064
		E1->O3	1	0.450	0.558	0.107	0.064
		I0->O0	1	0.199	0.225	0.107	0.064
		I1->O1	1	0.199	0.225	0.107	0.064
		I2->O2	1	0.199	0.225	0.107	0.064
		I3->O3	1	0.199	0.225	0.107	0.064
INV2	2x inverter (1)	I->O	2	0.238	0.168	0.030	0.026
INV2D	Dual 2x inverter (2)	I0->O0	2	0.238	0.168	0.030	0.026
		I1->O1	2	0.238	0.168	0.030	0.026
INV2T	2x Tri-state inverter (2)	I->O	2	0.345	0.305	0.048	0.036
		E->O	1.5	0.287	0.217	0.047	0.041

4

ATL 1.0 μ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

				Intercept		Slope	
Signal Name	Description (Site Count)	Path Timechk	Input Load	Rising Hold-R	Falling Hold-F	Rising Setup-R	Falling Setup-F
INV3	3x inverter (1)	I->O	3	0.227	0.154	0.022	0.020
INV4	4x inverter (1)	I->O	4	0.214	0.153	0.017	0.016
INV8	8x inverter (2)	I->O	8	0.178	0.113	0.010	0.010
INV10	10x inverter (4)	I->O	1	0.895	0.858	0.005	0.004
JKF	JK flip-flop (6)	CLK->Q	1	1.067	0.981	0.057	0.047
		J+CLK	2	0.000	0.000	0.641	0.626
		K+CLK	1	0.000	0.000	1.890	1.459
JKFBCPX	Clear preset JK flip-flop with asynchronous clear and preset and complementary outputs (8)	CLK->Q	1	1.185	1.339	0.061	0.051
		CLK->QB	1	1.849	1.616	0.061	0.051
		C->Q	2	-----	0.546	-----	0.050
		C->QB	2	1.280	-----	0.068	-----
		P->Q	2	1.263	-----	0.068	-----
		P->QB	2	-----	0.774	-----	0.051
		J+CLK	1	0.000	0.000	1.450	1.450
		K+CLK	1	0.000	0.000	0.850	1.450
JKFC	JK flip-flop with asynchronous clear (6)	CLK->Q	1	1.125	0.947	0.063	0.048
		C->Q	2	-----	0.970	-----	0.047
		J+CLK	2	0.000	0.000	0.815	0.610
		K+CLK	1	0.000	0.000	1.974	1.759
LAT	LATCH (3)	H->Q	2	0.904	0.730	0.057	0.046
		D->Q	1	0.711	0.817	0.057	0.047
		D+H		0.000	0.000	1.031	1.685
LATB	LATCH with complementary output (3)	H->Q	1	0.941	1.216	0.059	0.050
		H->QB	1	1.494	1.225	0.055	0.043
		D->Q	1	0.638	0.785	0.059	0.049
		D->QB	1	1.053	0.920	0.055	0.044
		D+H		0.000	0.000	0.990	0.785

ATL 1.0 µ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
LATBG	LATCH with complementary outputs and inverted gate signal (3)	G->Q	1	1.127	0.978	0.062	0.051
		G->QB	1	1.256	1.418	0.056	0.045
		D->Q	1	0.740	0.884	0.059	0.050
		D->QB	1	1.155	1.024	0.055	0.043
		D+G		0.000	0.000	0.690	1.400
LATBGQ	QUAD LATBG with common gate signal (9)	G->Q0	2	1.278	0.921	0.059	0.052
		G->Q1	2	1.278	0.921	0.059	0.052
		G->Q2	2	1.278	0.921	0.059	0.052
		G->Q3	2	1.278	0.921	0.059	0.052
		G->Q0B	2	1.139	1.517	0.058	0.043
		G->Q1B	2	1.139	1.517	0.058	0.043
		G->Q2B	2	1.139	1.517	0.058	0.043
		G->Q3B	2	1.139	1.517	0.058	0.043
		D0->Q0	1	0.641	0.790	0.059	0.048
		D1->Q1	1	0.641	0.790	0.059	0.048
		D2->Q2	1	0.641	0.790	0.059	0.048
		D3->Q3	1	0.641	0.790	0.059	0.048
		D0->Q0B	1	1.088	0.888	0.053	0.042
		D1->Q1B	1	1.088	0.888	0.053	0.042
		D2->Q2B	1	1.088	0.888	0.053	0.042
		D3->Q3B	1	1.088	0.888	0.053	0.042
		D0+G		0.069	0.000	0.371	1.085
		D1+G		0.069	0.000	0.371	1.085
		D2+G		0.069	0.000	0.371	1.085
		D3+G		0.069	0.000	0.371	1.085
LATBH	LATCH with high drive complementary outputs (4)	H->Q	1	0.930	1.125	0.032	0.033
		H->QB	1	1.498	1.266	0.027	0.022
		D->Q	1	0.689	0.823	0.032	0.030
		D->QB	1	1.182	1.027	0.024	0.022
		D+H		0.000	0.000	1.290	1.400
LATIQ	Quad inverting LATCH (7)	D0->Q0	2	0.414	0.347	0.054	0.046
		D1->Q1	2	0.414	0.347	0.054	0.046
		D2->Q2	2	0.414	0.347	0.054	0.046
		D3->Q3	2	0.414	0.347	0.054	0.046
		H->Q0	1	1.009	1.233	0.055	0.043
		H->Q1	1	1.009	1.233	0.055	0.043
		H->Q2	1	1.009	1.233	0.055	0.043
		H->Q3	1	1.009	1.233	0.055	0.043
		D0+H		0.715	0.270	0.294	0.151
		D1+H		0.715	0.270	0.294	0.151
		D2+H		0.715	0.270	0.294	0.151
		D3+H		0.715	0.270	0.294	0.151

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
LATR	LATCH with reset (3)	D->Q	1	0.843	0.875	0.102	0.047
		H->Q	2	1.012	0.794	0.102	0.047
		R->Q	1	-----	0.352	-----	0.043
		D+H		0.000	0.000	1.015	1.800
LATS	LATCH with set (3)	D->Q	1	0.751	1.169	0.058	0.053
		H->Q	2	0.911	0.961	0.058	0.055
		S->Q	2	0.839	-----	0.058	-----
		D+H		0.000	0.000	1.090	2.700
LATSR	LATCH with set and reset (3)	D->Q	1	0.876	1.166	0.102	0.054
		H->Q	2	1.019	0.971	0.102	0.056
		R->Q	1	-----	0.352	-----	0.042
		S->Q	2	0.864	-----	0.105	-----
		D+H		0.000	0.000	1.090	2.600
MUX2	2:1 MUX (2)	I0->O	1	0.664	0.786	0.058	0.049
		I1->O	1	0.664	0.786	0.058	0.049
		S->O	2	0.611	0.613	0.058	0.048
MUX2H	2:1 MUX - high drive (2)	I0->O	1	0.651	0.770	0.029	0.027
		I1->O	1	0.651	0.770	0.029	0.027
		S->O	2	0.843	0.874	0.029	0.026
MUX2I	2:1 MUX with inverted output (2)	I0->O	2	0.391	0.305	0.072	0.049
		I1->O	2	0.382	0.313	0.072	0.050
		S->O	2	0.612	0.548	0.075	0.043
MUX2IH	2:1 MUX with inverted output - high drive (2)	I0->O	3	0.371	0.258	0.038	0.032
		I1->O	3	0.349	0.269	0.039	0.033
		S->O	2	0.638	0.528	0.035	0.026
MUX2N	2:1 MUX with active low enable (2)	I0->O	1	0.742	0.806	0.103	0.049
		I1->O	1	0.729	0.800	0.103	0.050
		S->O	2	0.688	0.636	0.103	0.048
		E->O	1	0.355	0.298	0.099	0.044

ATL 1.0 μ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
MUX2NQ	Quad 2:1 MUX with active low enable (8)	IA0->OA	1	0.773	0.907	0.104	0.048
		IA1->OA	1	0.786	0.883	0.103	0.048
		IB0->OB	1	0.773	0.907	0.104	0.048
		IB1->OB	1	0.786	0.883	0.103	0.048
		IC0->OC	1	0.773	0.907	0.104	0.048
		IC1->OC	1	0.786	0.883	0.103	0.048
		ID0->OD	1	0.773	0.907	0.104	0.048
		ID1->OD	1	0.786	0.883	0.103	0.048
		E->OA	1	0.813	0.767	0.105	0.047
		E->OB	1	0.813	0.767	0.105	0.047
		E->OC	1	0.813	0.767	0.105	0.047
		E->OD	1	0.813	0.767	0.105	0.047
		S->OA	1	1.194	1.139	0.106	0.049
		S->OB	1	1.194	1.139	0.106	0.049
		S->OC	1	1.194	1.139	0.106	0.049
		S->OD	1	1.194	1.139	0.106	0.049
MUX2Q	Quad 2:1 MUX (6)	IA0->OA	1	0.656	0.806	0.058	0.050
		IA1->OA	1	0.667	0.792	0.057	0.048
		IB0->OB	1	0.656	0.806	0.058	0.050
		IB1->OB	1	0.667	0.792	0.057	0.048
		IC0->OC	1	0.656	0.806	0.058	0.050
		IC1->OC	1	0.667	0.792	0.057	0.048
		ID0->OD	1	0.656	0.806	0.058	0.050
		ID1->OD	1	0.667	0.792	0.057	0.048
		S->OA	1	1.059	1.065	0.056	0.050
		S->OB	1	1.059	1.065	0.056	0.050
		S->OC	1	1.059	1.065	0.056	0.050
		S->OD	1	1.059	1.065	0.056	0.050
MUX3I	3:1 MUX with inverted output (3)	I0->O	2	0.545	0.388	0.103	0.063
		I1->O	2	0.558	0.381	0.100	0.063
		I2->O	1	0.358	0.435	0.104	0.062
		S0->O	2	0.649	0.619	0.104	0.063
		S1->O	2	0.345	0.314	0.104	0.062
MUX3IH	3:1 MUX with inverted output - high drive (4)	I0->O	2	0.547	0.377	0.047	0.033
		I1->O	2	0.559	0.367	0.047	0.033
		I2->O	2	0.354	0.433	0.038	0.032
		S0->O	2	0.624	0.602	0.049	0.031
		S1->O	2	0.362	0.310	0.048	0.031

4

ATL 1.0 µ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
MUX4	4:1 MUX (4)	I0->O	1	1.091	1.324	0.065	0.059
		I1->O	1	1.091	1.324	0.065	0.059
		I2->O	1	1.091	1.324	0.065	0.059
		I3->O	1	1.091	1.324	0.065	0.059
		S0->O	3	1.135	1.385	0.066	0.061
		S1->O	2	1.078	0.815	0.068	0.060
MUX4X	4:1 MUX with transmission gate data inputs (5)	I0->O	2	0.645	0.758	0.050	0.043
		I1->O	2	0.645	0.758	0.050	0.043
		I2->O	2	0.645	0.758	0.050	0.043
		I3->O	2	0.645	0.758	0.050	0.043
		S0->O	3	0.953	1.050	0.051	0.047
		S1->O	2	0.558	0.582	0.048	0.043
MUX4XH	4:1 MUX with transmission gate data inputs - high drive (5)	I0->O	2	0.680	0.789	0.028	0.026
		I1->O	2	0.680	0.789	0.028	0.026
		I2->O	2	0.680	0.789	0.028	0.026
		I3->O	2	0.680	0.789	0.028	0.026
		S0->O	3	0.953	1.074	0.028	0.029
		S1->O	2	0.580	0.598	0.027	0.027
MUX5H	5:1 MUX - high drive (7)	I0->O	1	1.465	1.557	0.030	0.027
		I1->O	1	1.441	1.551	0.030	0.027
		I2->O	1	1.451	1.554	0.029	0.027
		I3->O	1	1.491	1.546	0.030	0.027
		I4->O	1	0.652	0.747	0.029	0.029
		S0->O	1	1.770	2.132	0.031	0.027
		S1->O	2	0.995	1.018	0.033	0.027
		S2->O	2	0.839	0.853	0.030	0.027
MUX8	8:1 MUX (10)	I0->O	1	1.588	1.862	0.072	0.070
		I1->O	1	1.588	1.862	0.072	0.070
		I2->O	1	1.588	1.862	0.072	0.070
		I3->O	1	1.588	1.862	0.072	0.070
		I4->O	1	1.588	1.862	0.072	0.070
		I5->O	1	1.588	1.862	0.072	0.070
		I6->O	1	1.588	1.862	0.072	0.070
		I7->O	1	1.588	1.862	0.072	0.070
		S0->O	1	2.302	2.527	0.073	0.070
		S1->O	1	1.940	1.719	0.074	0.075
		S2->O	1	0.909	0.931	0.069	0.063

Signal Name	Description (Site Count)	Path Timechk	Input Load	Rising Hold-R	Falling Hold-F	Rising Setup-R	Falling Setup-F
MUX8N	8:1 MUX with active low enable (10)	I0->O	1	1.753	1.833	0.110	0.075
		I1->O	1	1.753	1.833	0.110	0.075
		I2->O	1	1.753	1.833	0.110	0.075
		I3->O	1	1.753	1.833	0.110	0.075
		I4->O	1	1.753	1.833	0.110	0.075
		I5->O	1	1.753	1.833	0.110	0.075
		I6->O	1	1.753	1.833	0.110	0.075
		I7->O	1	1.753	1.833	0.110	0.075
		S0->O	1	2.378	2.578	0.107	0.073
		S1->O	1	1.544	1.693	0.108	0.073
		S2->O	1	0.989	0.830	0.109	0.062
		E->O	1	0.349	0.290	0.102	0.044
MUX8XH	8:1 MUX with transmission gate data inputs - high drive (10)	I0->O	2	1.014	1.152	0.036	0.033
		I1->O	2	1.014	1.152	0.036	0.033
		I2->O	2	1.014	1.152	0.036	0.033
		I3->O	2	1.014	1.152	0.036	0.033
		I4->O	2	1.014	1.152	0.036	0.033
		I5->O	2	1.014	1.152	0.036	0.033
		I6->O	2	1.014	1.152	0.036	0.033
		I7->O	2	1.014	1.152	0.036	0.033
		S0->O	1	1.598	1.772	0.034	0.035
		S1->O	3	0.919	0.956	0.035	0.035
		S2->O	2	0.566	0.654	0.033	0.031
NAN2	2 input NAND (1)	A->O	1	0.290	0.223	0.056	0.063
		B->O	1	0.248	0.257	0.056	0.064
NAN2D	Dual 2 input NAND (2)	A0->O0	1	0.290	0.223	0.056	0.063
		B0->O0	1	0.248	0.257	0.056	0.064
		A1->O1	1	0.290	0.223	0.056	0.063
		B1->O1	1	0.248	0.257	0.056	0.064
NAN2H	2 input NAND - high drive (1)	A->O	2	0.281	0.217	0.030	0.032
		B->O	2	0.221	0.229	0.033	0.035
NAN3	3 input NAND (1)	A->O	1	0.394	0.403	0.051	0.080
		B->O	1	0.394	0.403	0.051	0.080
		C->O	1	0.394	0.403	0.051	0.080

4

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
NAN3H	3 input NAND - high drive (2)	A->O	2	0.377	0.318	0.029	0.041
		B->O	2	0.314	0.328	0.029	0.040
		C->O	2	0.246	0.324	0.031	0.039
NAN4	4 input NAND (1)	A->O	1	0.429	0.578	0.054	0.101
		B->O	1	0.429	0.578	0.054	0.101
		C->O	1	0.429	0.578	0.054	0.101
		D->O	1	0.429	0.578	0.054	0.101
NAN4H	4 input NAND - high drive (2)	A->O	2	0.339	0.444	0.029	0.050
		B->O	2	0.341	0.453	0.029	0.049
		C->O	2	0.354	0.443	0.028	0.050
		D->O	2	0.347	0.442	0.029	0.050
NAN5	5 input NAND (3)	A->O	1	0.772	1.007	0.054	0.045
		B->O	1	0.758	0.831	0.054	0.046
		C->O	1	0.753	1.009	0.053	0.046
		D->O	1	0.680	0.858	0.055	0.048
		E->O	1	0.726	1.035	0.050	0.044
NAN5H	5 input NAND - high drive (3)	A->O	1	0.818	1.062	0.027	0.027
		B->O	1	0.785	0.899	0.026	0.028
		C->O	1	0.791	1.081	0.026	0.028
		D->O	1	0.707	0.920	0.029	0.029
		E->O	1	0.754	1.089	0.027	0.027
NAN6	6 input NAND (3)	A->O	1	0.841	1.085	0.054	0.046
		B->O	1	0.801	0.995	0.053	0.047
		C->O	1	0.800	1.040	0.055	0.045
		D->O	1	0.734	1.010	0.055	0.046
		E->O	1	0.786	1.039	0.050	0.046
		F->O	1	0.680	1.016	0.056	0.046
NAN6H	6 input NAND - high drive (3)	A->O	1	0.879	1.068	0.027	0.029
		B->O	1	0.822	1.046	0.026	0.029
		C->O	1	0.818	1.098	0.030	0.028
		D->O	1	0.789	1.073	0.027	0.028
		E->O	1	0.775	1.132	0.028	0.025
		F->O	1	0.775	1.076	0.023	0.029

ATL 1.0 µ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
NAN8	8 input NAND (3)	A->O	1	0.813	1.060	0.054	0.045
		B->O	1	0.788	1.039	0.052	0.047
		C->O	1	0.805	1.061	0.054	0.045
		D->O	1	0.761	1.052	0.053	0.046
		E->O	1	0.761	1.021	0.054	0.047
		F->O	1	0.714	1.032	0.053	0.047
		G->O	1	0.722	1.036	0.052	0.045
		H->O	1	0.656	1.021	0.053	0.046
NAN8H	8 input NAND - high drive (3)	A->O	1	0.911	1.211	0.026	0.029
		B->O	1	0.844	1.203	0.026	0.028
		C->O	1	0.867	1.223	0.029	0.026
		D->O	1	0.837	1.207	0.026	0.028
		E->O	1	0.853	1.222	0.026	0.027
		F->O	1	0.811	1.214	0.025	0.027
		G->O	1	0.804	1.206	0.026	0.026
		H->O	1	0.781	1.187	0.024	0.027
NOR2	2 input NOR (1)	A->O	1	0.345	0.290	0.100	0.044
		B->O	1	0.345	0.290	0.100	0.044
NOR2D	Dual 2 input NOR (2)	A0->O0	1	0.345	0.290	0.100	0.044
		B0->O0	1	0.345	0.290	0.100	0.044
		A1->O1	1	0.345	0.290	0.100	0.044
		B1->O1	1	0.345	0.290	0.100	0.044
NOR2H	2 input NOR - high drive (1)	A->O	2	0.297	0.211	0.046	0.027
		B->O	2	0.297	0.211	0.046	0.027
NOR3	3 input NOR (1)	A->O	1	0.477	0.312	0.151	0.044
		B->O	1	0.477	0.312	0.151	0.044
		C->O	1	0.477	0.312	0.151	0.044
NOR3H	3 input NOR - high drive (2)	A->O	2	0.447	0.270	0.070	0.026
		B->O	2	0.430	0.255	0.070	0.026
		C->O	2	0.390	0.168	0.062	0.028

4

ATL 1.0 μ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

				Intercept		Slope	
Signal Name	Description (Site Count)	Path Timechk	Input Load	Rising Hold-R	Falling Hold-F	Rising Setup-R	Falling Setup-F
NOR4	4 input NOR (1)	A->O	1	0.736	0.342	0.196	0.045
		B->O	1	0.736	0.342	0.196	0.045
		C->O	1	0.736	0.342	0.196	0.045
		D->O	1	0.736	0.342	0.196	0.045
NOR4H	4 input NOR - high drive (2)	A->O	2	0.580	0.262	0.094	0.026
		B->O	2	0.583	0.264	0.094	0.026
		C->O	2	0.605	0.304	0.094	0.024
		D->O	2	0.612	0.300	0.093	0.024
NOR5	5 input NOR (3)	A->O	1	1.007	0.714	0.057	0.043
		B->O	1	1.007	0.714	0.057	0.043
		C->O	1	1.007	0.714	0.057	0.043
		D->O	1	1.007	0.714	0.057	0.043
		E->O	1	1.007	0.714	0.057	0.043
NOR8	8 input NOR (3)	A->O	1	1.263	0.729	0.057	0.044
		B->O	1	1.263	0.729	0.057	0.044
		C->O	1	1.263	0.729	0.057	0.044
		D->O	1	1.263	0.729	0.057	0.044
		E->O	1	1.263	0.729	0.057	0.044
		F->O	1	1.263	0.729	0.057	0.044
		G->O	1	1.263	0.729	0.057	0.044
		H->O	1	1.263	0.729	0.057	0.044
NOR16H	16 input NOR - high drive (6)	A->O	1	1.724	0.866	0.037	0.026
		B->O	1	1.724	0.866	0.037	0.026
		C->O	1	1.724	0.866	0.037	0.026
		D->O	1	1.724	0.866	0.037	0.026
		E->O	1	1.724	0.866	0.037	0.026
		F->O	1	1.724	0.866	0.037	0.026
		G->O	1	1.724	0.866	0.037	0.026
		H->O	1	1.724	0.866	0.037	0.026
		I->O	1	1.724	0.866	0.037	0.026
		J->O	1	1.724	0.866	0.037	0.026
		K->O	1	1.724	0.866	0.037	0.026
		L->O	1	1.724	0.866	0.037	0.026
		M->O	1	1.724	0.866	0.037	0.026
		N->O	1	1.724	0.866	0.037	0.026
		O->O	1	1.724	0.866	0.037	0.026
		P->O	1	1.724	0.866	0.037	0.026

ATL 1.0 μ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
OAI22	2 input OR into 2 input NAND (1)	A->O	1	0.345	0.329	0.113	0.072
		B->O	1	0.462	0.211	0.117	0.078
		C->O	1	0.402	0.217	0.072	0.069
OAI22H	2 input OR into 3 input NAND - high drive (2)	A->O	2	0.345	0.329	0.057	0.036
		B->O	2	0.462	0.211	0.059	0.039
		C->O	2	0.402	0.217	0.036	0.034
OAI222	Two, 2 input ORs into 2 input NAND (2)	A->O	1	0.571	0.217	0.112	0.072
		B->O	1	0.468	0.292	0.111	0.071
		C->O	1	0.376	0.391	0.113	0.070
		D->O	1	0.449	0.275	0.122	0.076
OAI222H	Two, 2 input ORs into 2 input NAND - high drive (2)	A->O	2	0.571	0.217	0.056	0.036
		B->O	2	0.468	0.292	0.056	0.036
		C->O	2	0.376	0.391	0.056	0.035
		D->O	2	0.449	0.275	0.061	0.038
OAI22224	Four, 2 input ORs into 4 input NAND (3)	A->O	1	1.067	1.070	0.057	0.047
		B->O	1	1.067	1.070	0.057	0.047
		C->O	1	1.067	1.070	0.057	0.047
		D->O	1	1.067	1.070	0.057	0.047
		E->O	1	1.067	1.070	0.057	0.047
		F->O	1	1.067	1.070	0.057	0.047
		G->O	1	1.067	1.070	0.057	0.047
		H->O	1	1.067	1.070	0.057	0.047
OAI23	2 input OR into 3 input NAND (1)	A->O	1	0.464	0.555	0.099	0.083
		B->O	1	0.448	0.429	0.098	0.082
		C->O	1	0.473	0.433	0.055	0.069
		D->O	1	0.438	0.450	0.052	0.068
ORR2	2 input OR (1)	A->O	1	0.493	0.602	0.054	0.046
		B->O	1	0.397	0.618	0.054	0.045
ORR2H	2 input OR - high drive (1)	A->O	1	0.534	0.699	0.027	0.026
		B->O	1	0.534	0.699	0.027	0.026

4

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
ORR3	3 Input OR (2)	A->O	1	0.498	0.836	0.056	0.052
		B->O	1	0.498	0.836	0.056	0.052
		C->O	1	0.498	0.836	0.056	0.052
ORR3H	3 Input OR - high drive (2)	A->O	1	0.553	0.959	0.030	0.031
		B->O	1	0.553	0.959	0.030	0.031
		C->O	1	0.553	0.959	0.030	0.031
ORR4	4 Input OR (2)	A->O	1	0.566	1.217	0.059	0.059
		B->O	1	0.566	1.217	0.059	0.059
		C->O	1	0.566	1.217	0.059	0.059
		D->O	1	0.566	1.217	0.059	0.059
ORR4H	4 Input OR - high drive (2)	A->O	1	0.590	1.344	0.031	0.036
		B->O	1	0.590	1.344	0.031	0.036
		C->O	1	0.590	1.344	0.031	0.036
		D->O	1	0.590	1.344	0.031	0.036
ORR5	5 Input OR (2)	A->O	1	0.588	0.701	0.054	0.088
		B->O	1	0.588	0.701	0.054	0.088
		C->O	1	0.588	0.701	0.054	0.088
		D->O	1	0.588	0.701	0.054	0.088
		E->O	1	0.588	0.701	0.054	0.088
XNR2	2 Input Exclusive NOR (2)	A->O	2	0.456	0.508	0.100	0.061
		B->O	2	0.356	0.293	0.099	0.062
XNR2H	2 Input Exclusive NOR - high drive (2)	A->O	1	0.866	0.898	0.031	0.028
		B->O	2	0.821	0.681	0.031	0.027
XOR2	2 Input Exclusive OR (2)	A->O	2	0.693	0.640	0.099	0.065
		B->O	2	0.457	0.543	0.098	0.063
XOR2H	2 Input Exclusive OR - high drive (2)	A->O	1	0.698	0.713	0.030	0.030
		B->O	2	0.650	0.606	0.032	0.027

I/O Cells

Sample of buffers composed of modular I/O building blocks

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising	Intercept Falling	Slope Rising	Slope Falling
PADCB11T	2 mA bidi TTL input buffer	E->P		0.542	0.695	0.130	0.142
		AO->P		0.669	0.885	0.130	0.142
		P->AI		0.639	0.650	0.011	0.009
PADCB22C	4 mA bidi CMOS input buffer	E->P		0.668	0.633	0.065	0.072
		AO->P		0.821	0.947	0.065	0.071
		P->AI		0.884	0.920	0.005	0.005
PADCB22T	4 mA bidi TTL input buffer	E->P		0.976	1.135	0.058	0.079
		AO->P		0.969	1.116	0.060	0.079
		P->AI		0.639	0.650	0.011	0.009
PADCB22T260KU	4 mA bidi TTL buffer with pullup	E->P		0.976	1.135	0.058	0.079
		AO->P		0.969	1.116	0.060	0.079
		P->AI		0.639	0.650	0.011	0.009
PADCB22T40KU	4 mA bidi TTL buffer with pullup	E->P		0.976	1.135	0.058	0.079
		AO->P		0.969	1.116	0.060	0.079
		P->AI		0.639	0.650	0.011	0.009
PADCB33T	6 mA bidi TTL buffer	E->P		0.988	1.119	0.040	0.053
		AO->P		1.018	1.145	0.040	0.053
		P->AI		0.639	0.650	0.011	0.009
PADCB33T260KU	6 mA bidi TTL buffer with pullup	E->P		0.988	1.119	0.040	0.053
		AO->P		1.018	1.145	0.040	0.053
		P->AI		0.639	0.650	0.011	0.009
PADCB44T	8 mA bidi TTL buffer	E->P		1.004	1.178	0.030	0.040
		AO->P		1.015	1.119	0.030	0.040
		P->AI		0.639	0.650	0.011	0.009
PADCB44T40KU	8 mA bidi TTL buffer with pullup	E->P		1.004	1.178	0.030	0.040
		AO->P		1.015	1.119	0.030	0.040
		P->AI		0.639	0.650	0.011	0.009

I/O Cells

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising	Intercept Falling	Slope Rising	Slope Falling
PADCB55T	10 mA bidi TTL buffer	E->P		1.268	0.984	0.027	0.030
		AO->P		1.403	1.591	0.027	0.027
		P->AI		0.960	1.132	0.009	0.009
PADCB66T	12 mA bidi TTL buffer	E->P		1.060	1.176	0.020	0.027
		AO->P		1.040	1.095	0.021	0.027
		P->AI		0.639	0.650	0.011	0.009
PADCB66T40KU	12 mA bidi TTL buffer with pullup	E->P		1.060	1.176	0.020	0.027
		AO->P		1.040	1.095	0.021	0.027
		P->AI		0.639	0.650	0.011	0.009
PADCBCCTXLI	24 mA oscillator - 2 sites *(unique buffer please contact Atmel)*	E->P		1.962	0.861	0.015	0.013
		AO->P		2.172	1.600	0.015	0.012
		P->AI		0.444	0.517	0.008	0.007
PADCBCCTXRI	24 mA oscillator buffer - 2 sites *(unique buffer please contact Atmel)*	E->P		1.962	0.861	0.015	0.013
		AO->P		2.172	1.600	0.015	0.012
		P->AI		0.444	0.517	0.008	0.007
PADCIC	CMOS input buffer	P->AI		0.885	0.932	0.005	0.005
PADCIC100KU	CMOS input buffer with pullup	P->AI		1.501	1.720	0.010	0.008
PADCIC120KD	CMOS input buffer with pulldown	P->AI		1.627	1.574	0.010	0.009
PADCIC200KU	CMOS input buffer with pullup	P->AI		1.538	1.716	0.010	0.008
PADCICI	Inverting CMOS input buffer	P->AI		1.216	1.021	0.017	0.013
PADCIT	TTL input buffer	P->AI		0.639	0.650	0.011	0.009
PADCIT260KU	TTL input buffer with pullup	P->AI		0.639	0.650	0.011	0.009
PADCO11	2 mA output buffer	AO->P		0.625	1.260	0.130	0.136

I/O Cells

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising	Intercept Falling	Slope Rising	Slope Falling
PADCO22	4 mA output buffer	AO->P		0.969	1.116	0.060	0.079
PADCO33	6 mA output buffer	AO->P		1.018	1.145	0.040	0.053
PADCO44	8 mA output buffer	AO->P		1.015	1.119	0.030	0.040
PADCO66	12 mA output buffer	AO->P		1.040	1.095	0.021	0.027
PADCO88	16 mA output buffer	AO->P		1.852	1.710	0.018	0.017
PADCT22	4 mA tri state output buffer	E->P AO->P		0.657 0.810	0.621 0.943	0.065 0.065	0.072 0.071
PADCT33	6 mA Tri-state output buffer	E->P AO->P		0.988 1.018	1.119 1.145	0.040 0.040	0.053 0.053
PADCT44	8 mA Tri-state output buffer	E->P AO->P		1.004 1.015	1.178 1.119	0.030 0.030	0.040 0.040
PADCT55	10 mA Tri-state output buffer	E->P AO->P		1.257 1.396	0.971 1.580	0.027 0.027	0.030 0.027
PADCT66	12 mA Tri-state output buffer	E->P AO->P		1.060 1.040	1.176 1.095	0.020 0.021	0.027 0.027
PADCTCCXLI	24 mA Tri-state output buffer - 2 sites	E->P AO->P		1.962 2.172	0.861 1.600	0.015 0.015	0.013 0.012
PADCTCCXRI	24 mA Tri-state output buffer - 2 sites	E->P AO->P		1.962 2.172	0.861 1.600	0.015 0.015	0.013 0.012
PADCX22C	4 mA oscillator buffer - 2 sites	PI->AI PI->PO		0.959 0.295	1.077 0.277	0.010 0.068	0.010 0.075

4

Name	Description (Site Count)	Timechk	Load	Rising	Falling	Rising	Falling
PDB11T	2 mA bidi TTL buffer	E0->P		1.480	1.840	0.130	0.130
		E1->P		1.580	2.050	0.130	0.130
		AO->P		1.500	2.020	0.130	0.140
		P->AI		0.323	0.646	0.009	0.010
PDB22CN	4 mA bidi CMOS buffer with testable NAND	E0->P		1.203	1.687	0.130	0.073
		E1->P		1.177	1.887	0.130	0.073
		AO->P		1.157	1.979	0.130	0.072
		P->AI0		0.434	0.403	0.008	0.007
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDB22CU10N	4 mA bidi CMOS buffer with pullup and testable NAND	E0->P		1.203	1.687	0.130	0.073
		E1->P		1.177	1.887	0.130	0.073
		AO->P		1.157	1.979	0.130	0.072
		P->AI0		0.434	0.403	0.008	0.007
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDB2CN	4 mA bidi CMOS buffer with testable NAND	E0->P		1.554	0.772	0.065	0.073
		E1->P		1.523	1.040	0.065	0.073
		AO->P		1.601	1.218	0.065	0.071
		P->AI0		0.434	0.403	0.008	0.007
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDB2CU10N	4 mA bidi CMOS buffer with pullup and testable NAND	E0->P		1.554	0.772	0.065	0.073
		E1->P		1.523	1.040	0.065	0.073
		AO->P		1.601	1.218	0.065	0.071
		P->AI0		0.434	0.403	0.008	0.007
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDB2T	4 mA bidi TTL buffer	E0->P		1.554	0.772	0.065	0.073
		E1->P		1.523	1.040	0.065	0.073
		AO->P		1.601	1.218	0.065	0.071
		P->AI0		0.323	0.646	0.009	0.010

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising	Intercept Falling	Slope Rising	Slope Falling
PDB2TN	4 mA bidi TTL buffer with testable NAND	E0->P		1.554	0.772	0.065	0.073
		E1->P		1.523	1.040	0.065	0.073
		AO->P		1.601	1.218	0.065	0.071
		P->AI0		0.323	0.646	0.009	0.010
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDB2TU10N	4 mA bidi TTL buffer with pullup and testable NAND	E0->P		1.554	0.772	0.065	0.073
		E1->P		1.523	1.040	0.065	0.073
		AO->P		1.601	1.218	0.065	0.071
		P->AI0		0.323	0.646	0.009	0.010
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDB2TU2	4 mA bidi TTL buffer with pullup	E0->P		1.544	0.882	0.065	0.073
		E1->P		1.440	1.017	0.065	0.073
		AO->P		1.590	1.297	0.065	0.073
		P->AI0		0.765	1.222	0.010	0.007
PDB2TU4	4 mA bidi TTL buffer with pullup	E0->P		1.554	0.772	0.065	0.073
		E1->P		1.523	1.040	0.065	0.073
		AO->P		1.601	1.218	0.065	0.071
		P->AI0		0.323	0.646	0.009	0.010
PDB31CN	6 mA bidi CMOS buffer with testable NAND	E0->P		1.431	1.218	0.065	0.049
		E1->P		1.389	1.372	0.065	0.049
		AO->P		1.455	1.513	0.065	0.047
		P->AI0		0.434	0.403	0.008	0.007
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDB31CU10N	6 mA bidi CMOS buffer with pullup and testable NAND	E0->P		1.431	1.218	0.065	0.049
		E1->P		1.389	1.372	0.065	0.049
		AO->P		1.455	1.513	0.065	0.047
		P->AI0		0.434	0.403	0.008	0.007
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103

I/O Cells

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising	Intercept Falling	Slope Rising	Slope Falling
PDB32TU3SL	6 mA bidi TTL buffer with pullup and Schmitt Trigger	E0->P		5.419	3.831	0.070	0.055
		E1->P		5.289	3.942	0.070	0.054
		AO->P		5.645	4.323	0.070	0.056
		P->AI0		1.475	2.895	0.050	0.046
PDB3T	6 mA bidi TTL buffer	E0->P		1.665	0.862	0.044	0.049
		E1->P		1.628	1.131	0.043	0.049
		AO->P		1.720	1.260	0.043	0.049
		P->AI0		0.323	0.646	0.009	0.010
PDB3TN	6 mA bidi TTL buffer with testable NAND	E0->P		1.665	0.862	0.044	0.049
		E1->P		1.628	1.131	0.043	0.049
		AO->P		1.720	1.260	0.043	0.049
		P->AI0		0.323	0.646	0.009	0.010
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDB3TU10N	6 mA bidi TTL buffer with pullup and testable NAND	E0->P		1.665	0.862	0.044	0.049
		E1->P		1.628	1.131	0.043	0.049
		AO->P		1.720	1.260	0.043	0.049
		P->AI0		0.323	0.646	0.009	0.010
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDB41CSM	8 mA bidi CMOS buffer with Schmitt Trigger	E0->P		3.556	1.814	0.037	0.037
		E1->P		3.492	1.969	0.037	0.038
		AO->P		3.602	3.549	0.037	0.035
		P->AI0		1.987	1.426	0.036	0.060
PDB41CU10SM	8 mA bidi CMOS buffer with pullup and Schmitt Trigger	E0->P		3.556	1.814	0.037	0.037
		E1->P		3.492	1.969	0.037	0.038
		AO->P		3.602	3.549	0.037	0.035
		P->AI0		1.987	1.426	0.036	0.060
		E0->P		3.556	1.814	0.037	0.037
		E1->P		3.492	1.969	0.037	0.038
		AO->P		3.602	3.549	0.037	0.035
		P->AI0		0.729	1.317	0.050	0.033
PDB4CN	8 mA bidi CMOS buffer with testable NAND	E0->P		1.789	0.941	0.033	0.037
		E1->P		1.742	1.222	.033	0.037
		AO->P		1.775	1.372	.033	0.036
		P->AI0		0.434	0.403	0.008	0.007
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103

I/O Cells

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising	Intercept Falling	Slope Rising	Slope Falling
PDB4CU10	8 mA bidi CMOS buffer	E0->P		1.789	0.941	0.033	0.037
	with pullup	E1->P		1.742	1.222	0.033	0.037
		AO->P		1.775	1.372	0.033	0.036
		P->AI0		0.434	0.403	0.008	0.007
PDB4CU10N	8 mA bidi CMOS buffer with	E0->P		1.789	0.941	0.033	0.037
	pullup and testable NAND	E1->P		1.742	1.222	0.033	0.037
		AO->P		1.775	1.372	0.033	0.036
		P->AI0		0.434	0.403	0.008	0.007
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDB4T	8 mA bidi TTL buffer	E0->P		1.789	0.941	0.033	0.037
		E1->P		1.742	1.222	.033	0.037
		AO->P		1.775	1.372	0.033	0.036
		P->AI0		0.323	0.646	0.009	0.010
PDB4TN	8 mA bidi TTL buffer with	E0->P		1.789	0.941	0.033	0.037
	testable NAND	E1->P		1.742	1.222	0.033	0.037
		AO->P		1.775	1.372	0.033	0.036
		P->AI0		0.323	0.646	0.009	0.010
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDB4TU10	8 mA bidi TTL buffer with pullup	E0->P		1.789	0.941	0.033	0.037
		E1->P		1.742	1.222	0.033	0.037
		AO->P		1.775	1.372	0.033	0.036
		P->AI0		0.323	0.646	0.009	0.010
PDB4TU10N	8 mA bidi TTL buffer with	E0->P		1.789	0.941	0.033	0.037
	pullup and testable NAND	E1->P		1.742	1.222	0.033	0.037
		AO->P		1.775	1.372	0.033	0.036
		P->AI0		0.323	0.646	0.009	0.010
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDB4TU10SM	8 mA bidi TTL buffer with	E0->P		1.789	0.941	0.033	0.037
	pullup and Schmitt Trigger	E1->P		1.742	1.222	0.033	0.037
		AO->P		1.775	1.372	0.033	0.036
		P->AI0		0.729	1.317	0.050	0.033

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising	Intercept Falling	Slope Rising	Slope Falling
PDB51CU2SM	10 mA bidi CMOS buffer with pullup and Schmitt Trigger	E0->P		1.816	1.087	0.026	0.029
		E1->P		1.892	1.387	0.027	0.029
		AO->P		1.726	1.419	0.026	0.028
		P->AI0		2.036	1.348	0.033	0.066
PDB52CN	10 mA bidi CMOS buffer with testable NAND	E0->P		1.790	1.036	0.033	0.029
		E1->P		1.743	1.300	0.033	0.029
		AO->P		1.776	1.427	0.033	0.029
		P->AI0		0.434	0.403	0.008	0.007
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDB52CU10N	10 mA bidi CMOS buffer with pullup and testable NAND	E0->P		1.790	1.036	0.033	0.029
		E1->P		1.743	1.300	0.033	0.029
		AO->P		1.776	1.427	0.033	0.029
		P->AI0		0.434	0.403	0.008	0.007
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDB52TU10N	10 mA bidi TTL buffer with pullup and testable NAND	E0->P		1.790	1.036	0.033	0.029
		E1->P		1.743	1.300	0.033	0.029
		AO->P		1.776	1.427	0.033	0.029
		P->AI0		0.323	0.646	0.009	0.010
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDB53CU2SM	10 mA bidi CMOS buffer with pullup and Schmitt Trigger	E0->P		3.945	1.981	0.033	0.031
		E1->P		3.884	2.217	0.033	0.031
		AO->P		3.943	4.256	0.033	0.030
		P->AI0		2.036	1.348	0.033	0.066
PDB53TU10SM	10 mA bidi TTL buffer with pullup and Schmitt Trigger	E0->P		3.945	1.981	0.033	0.031
		E1->P		3.884	2.217	0.033	0.031
		AO->P		3.943	4.256	0.033	0.030
		P->AI0		0.729	1.317	0.050	0.033
PDB62CN	12 mA bidi CMOS buffer with testable NAND	E0->P		1.796	1.132	0.026	0.025
		E1->P		2.216	1.362	0.026	0.025
		AO->P		1.732	1.544	0.022	0.024
		P->AI0		0.434	0.403	0.008	0.007
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103

I/O Cells

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising	Intercept Falling	Slope Rising	Slope Falling
PDB62CU10N	12 mA bidi CMOS buffer with pullup and testable NAND	E0->P		1.796	1.132	0.026	0.025
		E1->P		2.216	1.362	0.026	0.025
		AO->P		1.732	1.544	0.022	0.024
		P->AI0		0.434	0.403	0.008	0.007
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDB62T	12 mA bidi TTL buffer	E0->P		1.796	1.132	.026	0.025
		E1->P		2.216	1.362	0.026	0.025
		AO->P		1.732	1.544	0.022	0.024
		P->AI0		0.323	0.646	0.009	0.010
PDB62TN	12 mA bidi TTL buffer with testable NAND	E0->P		1.796	1.132	0.026	0.025
		E1->P		2.216	1.362	0.026	0.025
		AO->P		1.732	1.544	0.022	0.024
		P->AI0		0.323	0.646	0.009	0.010
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDB62TU10N	12 mA bidi TTL buffer with pullup and testable NAND	E0->P		1.796	1.132	0.026	0.025
		E1->P		2.216	1.362	0.026	0.025
		AO->P		1.732	1.544	0.022	0.024
		P->AI0		0.323	0.646	0.009	0.010
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0 .087	0.103
PDB72CN	14 mA bidi CMOS buffer with testable NAND	E0->P		1.801	1.135	0.026	0.025
		E1->P		1.719	1.427	0.026	0.021
		AO->P		1.734	1.584	0.026	0.021
		P->AI0		0.434	0.403	0.008	0.007
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDB72CU10N	14 mA bidi CMOS buffer with pullup and testable NAND	E0->P		1.801	1.135	0.026	0.025
		E1->P		1.719	1.427	0.026	0.021
		AO->P		1.734	1.584	0.026	0.021
		P->AI0		0.434	0.403	0.008	0.007
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDBA1TU2SM	20 mA bidi TTL buffer with pullup and Schmitt Trigger	E0->P		3.363	1.685	0.020	0.018
		E1->P		3.488	2.003	0.020	0.018
		AO->P		3.420	2.501	0.020	0.016
		P->AI0		0.729	1.317	0.050	0.033

I/O Cells

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising	Intercept Falling	Slope Rising	Slope Falling
PDBC1TU4	24 mA bidi TTL buffer with pullup	E0->P		3.394	1.821	0.020	0.016
		E1->P		3.518	2.109	0.020	0.017
		AO->P		3.465	3.151	0.019	0.013
		P->AI0		0.323	0.646	0.009	0.010
PDBC1TU4S	24 mA bidi TTL buffer with pullup and Schmitt Trigger	E0->P		3.394	1.821	0.020	0.016
		E1->P		3.518	2.109	0.020	0.017
		AO->P		3.465	3.151	0.019	0.013
		P->AI0		1.799	2.907	0.010	0.010
PDBC1TU4SM	24 mA bidi TTL buffer with pullup and Schmitt Trigger	E0->P		3.394	1.821	0.020	0.016
		E1->P		3.518	2.109	0.020	0.017
		AO->P		3.465	3.151	0.019	0.013
		P->AI0		0.729	1.317	0.050	0.033
PDIC	CMOS input buffer	P->AI0		0.434	0.403	0.008	0.007
PDIC02	CMOS input buffer	P->AI0		0.434	0.403	0.008	0.007
PDIC02U10N	CMOS input buffer with pullup and testable NAND	P->AI0		0.434	0.403	0.008	0.007
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDICD10SM	CMOS input buffer with pulldown and Schmitt Trigger	P->AI0		1.987	1.426	0.036	0.060
PDICD15N	CMOS input buffer with pulldown and testable NAND	P->AI0		0.434	0.403	0.008	0.007
		AI0->AI1		0.119	0.200	0 .087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDICD15SN	CMOS input buffer with pulldown Schmitt Trigger and testable NAND	P->AI0		1.741	2.041	0.010	0.009
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103
PDICN	CMOS input buffer with testable NAND	P->AI0		0.434	0.403	0.008	0.007
		AI0->AI1		0.119	0.200	0.087	0.101
		E2->AI1		0.148	0.133	0.087	0.103

I/O Cells

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising	Intercept Falling	Slope Rising	Slope Falling
PDICNI	Inverting CMOS input buffer with testable NAND	P->AI0 AI0->AI1 E2->AI1		0.100 0.119 0.148	0.093 0.200 0.133	0 .021 0.087 0.087	0.025 0.101 0.103
PDICS	CMOS input buffer with Schmitt Trigger	P->AI0		1.741	2.041	0.010	0.009
PDICSM	CMOS input buffer with Schmitt Trigger	P->AI0		1.987	1.426	0.036	0.060
PDICSN	CMOS input buffer with Schmitt Trigger and testable NAND	P->AI0 AI0->AI1 E2->AI1		1.741 0.119 0.148	2.041 0.200 0.133	0.010 0.087 0.087	0.009 0.101 0.103
PDICSNI	Inverting CMOS input buffer with Schmitt Trigger and testable NAND	P->AI0 AI0->AI1 E2->AI1		1.187 0.119 0.148	1.182 0.200 0.133	0.013 0.087 0.087	0.015 0.101 0.103
PDICU10	CMOS input buffer with pullup	P->AI0		0.434	0.403	.008	0.007
PDICU10N	CMOS input buffer with pullup and testable NAND	P->AI0 AI0->AI1 E2->AI1		0.434 0.119 0.148	0.403 0.200 0.133	0.008 0.087 0.087	0.007 0.101 0.103
PDICU10NI	Inverting CMOS input buffer with pullup and testable NAND	P->AI0 AI0->AI1 E2->AI1		0.100 0.119 0.148	0.093 0.200 0.133	0.021 0.087 0.087	0.025 0.101 0.103
PDICU10SM	CMOS input buffer with pullup and Schmitt Trigger	P->AI0		1.987	1.426	0.036	0.060
PDICU2	CMOS input buffer with pullup	P->AI0		0.434	0.403	0.008	0.007
PDICU2SM	CMOS input buffer with pullup and Schmitt Trigger	P->AI0		1.987	1.426	0.036	0.060
PDICU31S	CMOS input buffer with pullup and Schmitt Trigger	P->AI0		1.741	2.041	0.010	0.009

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising	Intercept Falling	Slope Rising	Slope Falling
PDIT	TTL input buffer	P->AI0		0.323	0.646	0.009	0.010
PDITD15	TTL input buffer with pulldown	P->AI0		0.323	0.646	0.009	0.010
PDITN	TTL input buffer with testable NAND	P->AI0 AI0->AI1 E2->AI1		0.323 0.119 0.148	0.646 0.200 0.133	0.009 0.087 0.087	0.010 0.101 0.103
PDITS	TTL input buffer with Schmitt Trigger	P->AI0		1.799	2.907	0.010	0.010
PDITSM	TTL input buffer with Schmitt Trigger	P->AI0		0.729	1.317	0.050	0.033
PDITU10	TTL input buffer with pullup	P->AI0		0.323	0.646	0.009	0.010
PDITU10N	TTL input buffer with pullup and testable NAND	P->AI0 AI0->AI1 E2->AI1		0.323 0.119 0.148	0.646 0.200 0.133	0.009 0.087 0.087	0.010 0.101 0.103
PDITU10SM	TTL input buffer with pullup and Schmitt Trigger	P->AI0		0.729	1.317	0.050	0.033
PDITU1S	TTL input buffer with pullup and Schmitt Trigger	P->AI0		1.799	2.907	0.010	0.010
PDITU2N	TTL input buffer with pullup and testable NAND	P->AI0 AI0->AI1 E2->AI1		0.323 0.119 0.148	0.646 0.200 0.133	0.009 0.087 0.087	0.010 0.101 0.103
PDITU2SM	TTL input buffer with pullup and Schmitt Trigger	P->AI0		0.729	1.317	0.050	0.033
PDITU31	TTL input buffer with pullup	P->AI0		0.323	0.646	0.009	0.010
PDITU4	TTL input buffer with pullup	P->AI0		0.323	0.646	0.009	0.010

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising	Intercept Falling	Slope Rising	Slope Falling
PDITU4S	TTL input buffer with pullup	P->AI0		1.799	2.907	0.010	0.010
PDK3	6 mA clock driver	I->O		0.498	0.435	0.003	0.003
PDK4	8 mA clock driver	I->O		0.417	0.490	0.002	0.002
PDK6	12 mA clock driver	I->O		0.531	0.606	0.001	0.002
PDKC	24 mA clock driver	I->O		0.868	0.988	0.001	0.001
PDO1	2 mA output buffer	AO->P		0.460	1.010	0.130	0.140
PDO2	4 mA output buffer	AO->P		0.459	0.688	0.065	0.072
PDO21	4 mA output buffer	AO->P		1.429	1.179	0.065	0.071
PDO22	4 mA output buffer	AO->P		0.544	1.360	0.130	0.069
PDO23	4 mA output buffer	AO->P		4.780	4.850	0.073	0.082
PDO4	8 mA output buffer	AO->P		0.587	0.712	0.032	0.036
PDO51	10 mA output buffer	AO->P		1.037	1.085	0.027	0.027
PDO61	12 mA output buffer	AO->P		0.670	0.854	0.026	0.024
PDO71	14 mA output buffer	AO->P		0.755	0.867	0.026	0.021
PDOC1	24 mA output buffer	AO->P		2.989	2.778	0.019	0.013
PDT11	2 mA Tri-state output buffer	E0->P E1->P AO->P		1.453 1.578 1.437	2.803 3.140 2.500	0.130 0.130 0.130	0.100 0.100 0.121

4

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising	Intercept Falling	Slope Rising	Slope Falling
PDT2	4 mA Tri-state output buffer	E0->P		1.508	0.855	0.065	0.072
		E1->P		1.523	1.155	0.068	0.072
		AO->P		1.565	1.269	0.065	0.072
PDT22	4 mA Tri-state output buffer	E0->P		1.203	1.687	0.130	0.073
		E1->P		1.177	1.887	0.130	0.073
		AO->P		1.124	2.024	0.130	0.070
PDT3	6 mA Tri-state output buffer	E0->P		1.665	0.862	0.044	0.049
		E1->P		1.628	1.131	0.043	0.049
		AO->P		1.720	1.260	0.043	0.049
PDT31	6 mA Tri-state output buffer	E0->P		1.431	1.218	0.065	0.049
		E1->P		1.389	1.372	0.065	0.049
		AO->P		1.455	1.513	0.065	0.047
PDT4	8 mA Tri-state output buffer	E0->P		1.789	0.941	0.033	0.037
		E1->P		1.742	1.222	0.033	0.037
		AO->P		1.775	1.372	0.033	0.036
PDT41	8 mA Tri-state output buffer	E0->P		3.556	1.814	0.037	0.037
		E1->P		3.492	1.969	0.037	0.038
		AO->P		3.600	3.549	0.037	0.035
PDT41U10	8 mA Tri-state output buffer with pullup	E0->P		3.556	1.814	0.037	0.037
		E1->P		3.492	1.969	0.037	0.038
		AO->P		3.600	3.549	0.037	0.035
PDT4U10	8 mA Tri-state output buffer with pullup	E0->P		1.789	0.941	0.033	0.037
		E1->P		1.742	1.222	0.033	0.037
		AO->P		1.775	1.372	0.033	0.036
PDTC1	24 mA Tri-state output buffer	E0->P		3.394	1.821	0.020	0.016
		E1->P		3.518	2.109	0.020	0.017
		AO->P		3.465	3.151	0.019	0.013
PDX2CI	4 mA with inverting CMOS oscillator buffer	PI->AI1		0.609	0.497	0.111	0.143
		E2->AI1		0.599	0.524	0.110	0.149
		AI1->PO		0.377	0.521	0.065	0.071
		PO->AI0		0.098	0.112	0.028	0.028

I/O Cells

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising	Intercept Falling	Slope Rising	Slope Falling
PSB2C	4 mA bidi CMOS buffer	E0->P		1.027	0.932	0.065	0.072
		AO->P		0.923	1.093	0.065	0.072
		P->AI0		0.434	0.403	0.007	0.007
PSB2CS	4 mA bidi CMOS buffer with Schmitt Trigger	E0->P		1.027	0.932	0.065	0.072
		AO->P		0.923	1.093	0.065	0.072
		P->AI0		1.741	2.041	0.010	0.009
PSB2TU31	4 mA bidi TTL buffer with pullup	E0->P		1.027	0.932	0.065	0.072
		AO->P		0.923	1.093	0.065	0.072
		P->AI0		0.323	0.646	0.009	0.010
PSB91CS	18 mA bidi CMOS buffer with Schmitt Trigger	E0->P		2.313	1.729	0.044	0.019
		AO->P		2.291	3.162	0.044	0.021
		P->AI0		1.741	2.041	0.010	0.009
PSB91CU2SM	18 mA bidi CMOS buffer with Schmitt Trigger	E0->P		2.313	1.729	0.044	0.019
		AO->P		2.291	3.162	0.044	0.021
		P->AI0		2.036	1.348	0.033	0.066
PSB91TS	18 mA bidi TTL buffer with Schmitt Trigger	E0->P		2.313	1.729	0.044	0.019
		AO->P		2.291	3.162	0.044	0.021
		P->AI0		1.799	2.907	0.010	0.010
PSB91TU2SM	18 mA bidi TTL buffer with pullup and Schmitt Trigger	E0->P		2.313	1.729	0.044	0.019
		AO->P		2.291	3.162	0.044	0.021
		P->AI0		0.729	1.317	0.050	0.033
PSB9CS	18 mA bidi CMOS buffer with Schmitt Trigger	E0->P		1.976	1.120	0.015	0.017
		AO->P		1.840	1.560	0.015	0.016
		P->AI0		1.741	2.041	0.010	0.009
PSB9TS	18 mA bidi TTL buffer with Schmitt Trigger	E0->P		1.976	1.120	0.015	0.017
		AO->P		1.840	1.560	0.015	0.016
		P->AI0		1.799	2.907	0.010	0.010
PSBC1TS	24 mA bidi TTL buffer with Schmitt Trigger	E0->P		3.503	2.750	0.020	0.021
		AO->P		3.470	3.350	0.019	0.019
		P->AI0		1.799	2.907	0.010	0.010

I/O Cells

Signal Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising	Intercept Falling	Slope Rising	Slope Falling
PSBC2TS	24 mA bidi TTL buffer with Schmitt Trigger	E0->P		5.171	5.712	0.029	0.035
		AO->P		5.073	5.905	0.029	0.035
		P->AI0		1.799	2.907	0.010	0.010
PSBCT	24 mA bidi TTL buffer	E0->P		2.064	1.326	0.012	0.013
		AO->P		2.020	1.854	0.012	0.011
		P->AI0		0.323	0.646	0.009	0.010
PST2	4 mA Tri-state output buffer	E0->P		1.027	0.932	0.065	0.072
		AO->P		0.923	1.093	0.065	0.072
PST4	8 mA Tri-state output buffer	E0->P		1.392	1.112	0.033	0.037
		AO->P		1.271	1.419	0.033	0.035
PSTA	20 mA Tri-state output buffer	E0->P		1.878	1.093	0.014	0.015
		AO->P		1.915	1.933	0.014	0.014
PSTC	24 mA Tri-state output buffer	E0->P		2.064	1.326	0.012	0.013
		AO->P		2.020	1.854	0.012	0.011
PSTC1	24 mA Tri-state output buffer	E0->P		3.426	2.742	0.020	0.021
		AO->P		3.376	3.291	0.020	0.019
PSTC2	24 mA Tri-state output buffer	E0->P		5.153	5.695	0.029	0.035
		AO->P		5.058	5.888	0.029	0.035

ATL80 0.8 μ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Macrocells in alpha-numeric order

Cell Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
ADD3	1 bit full adder (12)	P->CO	1	0.989	1.012	0.069	0.039
		P->SO	1	0.953	0.939	0.066	0.038
		Q->CO	4	0.220	0.371	0.053	0.040
		Q->SO	4	0.362	0.403	0.068	0.039
		CI->CO	3	0.369	0.411	0.055	0.040
		CI->SO	3	0.244	0.365	0.065	0.038
ADD3X	1 bit full adder with buffered outputs (10)	P->CO	2	0.703	0.743	0.051	0.025
		P->SO	2	0.540	0.634	0.051	0.026
		Q->CO	2	0.424	0.862	0.048	0.026
		Q->SO	2	1.025	0.748	0.052	0.025
		CI->CO	2	0.444	0.501	0.050	0.025
		CI->SO	2	0.259	0.375	0.051	0.024
AND2	2 input AND (2)	A->O	1	0.261	0.331	0.052	0.023
		B->O	1	0.279	0.264	0.051	0.022
AND2H	2 input AND - high drive (3)	A->O	1	0.428	0.415	0.015	0.009
		B->O	1	0.456	0.359	0.015	0.009
AND3	3 input AND (3)	A->O	1	0.483	0.441	0.023	0.013
		B->O	1	0.508	0.389	0.023	0.013
		C->O	1	0.506	0.331	0.023	0.012
AND3H	3 input AND - high drive (4)	A->O	1	0.622	0.519	0.012	0.008
		B->O	1	0.644	0.470	0.012	0.008
		C->O	1	0.657	0.414	0.012	0.008
AND4	4 input AND (3)	A->O	1	0.537	0.460	0.051	0.024
		B->O	1	0.534	0.427	0.052	0.024
		C->O	1	0.542	0.372	0.051	0.023
		D->O	1	0.497	0.299	0.052	0.023
AND4H	4 input AND - high drive (4)	A->O	1	0.712	0.519	0.017	0.011
		B->O	1	0.727	0.495	0.016	0.010
		C->O	1	0.717	0.439	0.017	0.010
		D->O	1	0.702	0.389	0.016	0.009

ATL80 0.8 μ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Cell Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
AND5	5 input AND (5)	A->O	1	0.402	0.418	0.095	0.023
		B->O	1	0.317	0.367	0.095	0.023
		C->O	1	0.409	0.364	0.095	0.023
		D->O	1	0.322	0.285	0.095	0.023
		E->O	1	0.412	0.291	0.095	0.022
AOI22	2 input AND into 2 input NOR (2)	A->O	1	0.172	0.208	0.095	0.035
		B->O	1	0.127	0.248	0.095	0.037
		C->O	1	0.150	0.236	0.095	0.026
AOI22H	2 input AND into 2 input NOR - high drive (4)	A->O	2	0.275	0.200	0.045	0.020
		B->O	2	0.180	0.225	0.046	0.021
		C->O	2	0.208	0.207	0.046	0.015
AOI222	Two, 2 input ANDs into 2 input NOR (4)	A->O	1	0.162	0.355	0.072	0.037
		B->O	1	0.245	0.318	0.071	0.036
		C->O	1	0.178	0.222	0.072	0.036
		D->O	1	0.119	0.258	0.073	0.037
AOI222H	Two, 2 input ANDs into 2 input NOR - high drive (8)	A->O	2	0.223	0.343	0.032	0.020
		B->O	2	0.290	0.306	0.032	0.019
		C->O	2	0.236	0.209	0.033	0.019
		D->O	2	0.169	0.235	0.033	0.021
AOI2223	Three, 2 input ANDs into 3 input NOR (4)	A->O	1	0.392	0.383	0.091	0.036
		B->O	1	0.308	0.425	0.092	0.037
		C->O	1	0.320	0.327	0.092	0.036
		D->O	1	0.245	0.362	0.092	0.037
		E->O	1	0.224	0.237	0.093	0.036
		F->O	1	0.143	0.268	0.093	0.037
AOI2223H	Three, 2 input ANDs into 3 input NOR - high drive (7)	A->O	1	0.578	0.619	0.023	0.013
		B->O	1	0.645	0.634	0.023	0.013
		C->O	1	0.553	0.591	0.023	0.013
		D->O	1	0.631	0.587	0.023	0.013
		E->O	1	0.521	0.532	0.023	0.012
		F->O	1	0.608	0.524	0.023	0.012
AOI23	2 input AND into 3 input NOR (2)	A->O	1	0.335	0.228	0.136	0.036
		B->O	1	0.374	0.146	0.152	0.033
		C->O	1	0.499	0.137	0.152	0.024
		D->O	1	0.463	0.126	0.152	0.024

Cell Name	Description (Site Count)	Path Timechk	Input Load	Rising Hold-R	Falling Hold-F	Rising Setup-R	Falling Setup-F
BUF1	1x buffer (2)	I->O	1	0.164	0.237	0.050	0.022
BUF2	2x buffer (2)	I->O	1	0.257	0.291	0.022	0.012
BUF2T	2x Tri State bus driver with active high enable (4)	I->O	1	0.417	0.307	0.052	0.032
		E->O	1.5	0.234	0.105	0.052	0.033
BUF2Z	2x Tri State bus driver with active low enable (4)	I->O	1	0.314	0.333	0.072	0.028
		E->O	1.5	0.057	0.216	0.074	0.028
BUF3	3x buffer (3)	I->O	1	0.318	0.344	0.014	0.009
BUF4	4x buffer (3)	I->O	1	0.374	0.393	0.010	0.007
BUF8	8x buffer (5)	I->O	2	0.378	0.388	0.006	0.005
BUF12	12x buffer (8)	I->O	3	0.399	0.418	0.004	0.002
BUF16	16x buffer (10)	I->O	4	0.406	0.416	0.003	0.002
CLA7X	7 input carry lookahead (5)	A->O	1	1.051	0.551	0.177	0.062
		B->O	1	1.193	0.623	0.177	0.063
		C->O	1	1.089	0.458	0.177	0.049
		D->O	1	0.581	0.447	0.136	0.049
		E->O	1	0.504	0.310	0.136	0.036
		F->O	1	0.174	0.316	0.095	0.037
		G->O	1	0.117	0.182	0.095	0.027
DEC4	2:4 decoder (7)	S0->D0	3	0.233	0.357	0.051	0.034
		S0->D1	3	0.109	0.278	0.052	0.038
		S0->D2	3	0.233	0.357	0.051	0.034
		S0->D3	3	0.109	0.278	0.052	0.038
		S1->D0	3	0.269	0.368	0.052	0.034
		S1->D1	3	0.271	0.365	0.051	0.034
		S1->D2	3	0.168	0.232	0.050	0.036
		S1->D3	3	0.170	0.236	0.050	0.036

4

Cell Name	Description (Site Count)	Path Timechk	Input Load	Rising Hold-R	Falling Hold-F	Rising Setup-R	Falling Setup-F
DEC4N	2:4 decoder with active low enable (9)	S0->D0	1	0.321	0.612	0.050	0.035
		S0->D1	1	0.636	0.703	0.050	0.035
		S0->D2	1	0.280	0.634	0.051	0.037
		S0->D3	1	0.667	0.701	0.051	0.035
		S1->D0	1	0.273	0.392	0.052	0.035
		S1->D1	1	0.329	0.396	0.051	0.035
		S1->D2	1	0.456	0.529	0.050	0.035
		S1->D3	1	0.462	0.516	0.043	0.035
		E->D0	2	0.374	0.578	0.049	0.035
		E->D1	2	0.280	0.536	0.052	0.035
		E->D2	2	0.319	0.590	0.052	0.036
		E->D3	2	0.318	0.527	0.052	0.035
DEC8N	3:8 decoder with active low enable (24)	S0->D0	1	0.502	0.872	0.052	0.047
		S0->D1	1	1.029	1.069	0.051	0.047
		S0->D2	1	0.502	0.872	0.052	0.047
		S0->D3	1	1.029	1.069	0.051	0.047
		S0->D4	1	0.502	0.872	0.052	0.047
		S0->D5	1	1.029	1.069	0.051	0.047
		S0->D6	1	0.502	0.872	0.052	0.047
		S0->D7	1	1.029	1.069	0.051	0.047
		S1->D0	1	0.456	0.630	0.050	0.047
		S1->D1	1	0.456	0.630	0.050	0.047
		S1->D2	1	0.697	0.874	0.052	0.047
		S1->D3	1	0.697	0.874	0.052	0.047
		S1->D4	1	0.456	0.630	0.050	0.047
		S1->D5	1	0.456	0.630	0.050	0.047
		S1->D6	1	0.697	0.874	0.052	0.047
		S1->D7	1	0.697	0.874	0.052	0.047
		S2->D0	1	0.482	0.631	0.051	0.047
		S2->D1	1	0.482	0.631	0.051	0.047
		S2->D2	1	0.482	0.631	0.051	0.047
		S2->D3	1	0.482	0.631	0.051	0.047
		S2->D4	1	0.741	0.878	0.051	0.047
		S2->D5	1	0.741	0.878	0.051	0.047
		S2->D6	1	0.741	0.878	0.051	0.047
		S2->D7	1	0.741	0.878	0.051	0.047
		E->D0	2	0.553	0.840	0.051	0.047
		E->D1	2	0.553	0.840	0.051	0.047
		E->D2	2	0.553	0.840	0.051	0.047
		E->D3	2	0.553	0.840	0.051	0.047
		E->D4	2	0.553	0.840	0.051	0.047
		E->D5	2	0.553	0.840	0.051	0.047
		E->D6	2	0.553	0.840	0.051	0.047
		E->D7	2	0.553	0.840	0.051	0.047
DFF	D flip-flop (8)	CLK->Q	1	0.646	0.612	0.023	0.017
		D+CLK		0.048	0.079	0.102	0.211

ATL80 0.8 μ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Cell Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
DFFBCPX	D flip-flop with asynchronous clear and preset with complementary outputs (16)	C->Q	2	1.687	0.656	0.015	0.015
		C->QB	2	1.343	0.733	0.010	0.014
		CLK->Q	1	1.142	0.906	0.022	0.012
		CLK->QB	1	1.230	1.378	0.022	0.013
		P->Q	2	1.687	0.656	0.015	0.015
		P->QB	2	1.343	0.733	0.010	0.014
		D+CLK		0.000	0.015	0.258	0.222
DFFBSRX	D flip-flop with asynchronous set and reset with complementary outputs (16)	CLK->Q	1	0.806	0.929	0.023	0.012
		CLK->QB	1	1.105	1.086	0.022	0.012
		R->Q	2	0.583	0.975	0.016	0.011
		R->QB	2	0.658	0.991	0.016	0.011
		S->Q	2	0.583	0.975	0.016	0.011
		S->QB	2	0.658	0.991	0.016	0.011
		D+CLK		0.158	0.056	0.102	0.399
DFFC	D flip-flop with asynchronous clear (9)	CLK->Q	1	0.693	0.590	0.027	0.019
		C->Q	2	——	0.674	——	0.021
		D+CLK		0.000	0.056	0.197	0.211
DFFH	D flip-flop - high drive (8)	CLK->Q	1	0.647	0.640	0.023	0.016
		D+CLK		0.048	0.079	0.102	0.211
DFFR	D flip-flop with asynchronous reset (11)	CLK->Q	1	0.677	0.589	0.017	0.012
		R->Q	1	——	0.916	——	0.011
		D+CLK		0.000	0.173	0.197	0.192
DFFS	D flip-flop with asynchronous set (9)	CLK->Q	1	0.687	0.701	0.024	0.018
		S->Q	2	0.497	——	0.024	——
		D+CLK		0.079	0.000	0.081	0.740
DFFSR	D flip-flop with asynchronous set and reset (12)	CLK->Q	1	0.947	0.847	0.023	0.012
		S->Q	2	0.854	——	0.023	——
		R->Q	2	0.854	0.505	0.023	0.012
		D+CLK		0.056	0.056	0.102	0.506
DLY1500	Delay buffer 1.5 ns (6)	I->O	1	1.451	1.406	0.052	0.027
DLY2100	Delay buffer 2.1 ns (10)	I->O	1	2.289	2.108	0.050	0.025
DLY6000	Delay buffer 6.0 ns (24)	I->O	1	6.178	5.759	0.052	0.027

4

ATL80 0.8 μ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Cell Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
DSS	Set scan D flip-flop (11)	CLK->Q	1	0.776	0.904	0.016	0.009
		D+CLK		0.000	0.000	0.554	0.697
		TI+CLK		0.000	0.000	0.625	0.613
		TE+CLK		0.000	0.000	0.762	0.730
DSSR	Set scan D flip-flop with reset (13)	CLK->Q	1	0.704	0.779	0.014	0.009
		R->Q	2	—	0.658	—	0.009
		D+CLK		0.000	0.000	0.855	0.697
		TI+CLK		0.000	0.000	0.825	0.707
		TE+CLK		0.000	0.000	0.966	0.730
INV1	1x inverter (1)	I->O	1	0.084	0.174	0.051	0.026
INV1D	Dual 1x inverters (2)	I0->O0	1	0.084	0.174	0.051	0.026
		I1->O1	1	0.084	0.174	0.051	0.026
INV1Q	Quad 1x inverters (4)	I0->O0	1	0.084	0.174	0.051	0.026
		I1->O1	1	0.084	0.174	0.051	0.026
		I2->O2	1	0.084	0.174	0.051	0.026
		I3->O3	1	0.084	0.174	0.051	0.026
INV1TQ	Quad Tri State inverter (7)	E0->O0	1	0.471	0.729	0.091	0.036
		E0->O1	1	0.471	0.729	0.091	0.036
		E0->O2	1	0.471	0.729	0.091	0.036
		E0->O3	1	0.471	0.729	0.091	0.036
		E1->O0	1	0.454	0.698	0.092	0.038
		E1->O1	1	0.454	0.698	0.092	0.038
		E1->O2	1	0.454	0.698	0.092	0.038
		E1->O3	1	0.454	0.698	0.092	0.038
		I0->O0	1	0.212	0.277	0.094	0.036
		I1->O1	1	0.212	0.277	0.094	0.036
		I2->O2	1	0.212	0.277	0.094	0.036
		I3->O3	1	0.212	0.277	0.094	0.036
INV2	2x inverter (2)	I->O	2	0.113	0.137	0.023	0.016
INV2D	Dual 2x inverter (4)	I0->O0	2	0.113	0.137	0.023	0.016
		I1->O1	2	0.113	0.137	0.023	0.016

ATL80 0.8 μ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Cell Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
INV2T	2x Tri State inverter with active high enable (3)	I->O	2	0.191	0.102	0.075	0.028
		E->O	1.5	0.124	0.105	0.073	0.033
INV3	3x inverter (2)	I->O	3	0.124	0.123	0.016	0.012
INV4	4x inverter (2)	I->O	4	0.138	0.084	0.013	0.011
INV8	8x inverter (4)	I->O	8	0.172	0.083	0.006	0.005
INV10	10x inverter (8)	I->O	1	0.528	0.562	0.004	0.003
JKF	JK flip flop (10)	CLK->Q		0.658	0.617	0.023	0.014
		J+CLK		0.000	0.000	0.435	0.427
		K+CLK		0.000	0.000	0.806	0.687
JKFC	JK flip flop with asynchronous clear (12)	CLK->Q		0.727	0.615	0.048	0.025
		C->Q		1.322	0.584	0.048	0.025
		J+CLK		0.000	0.000	0.743	0.382
		K+CLK		0.000	0.000	0.894	0.694
LAT	LATCH (4)	H->Q	2	0.414	0.517	0.052	0.025
		D->Q	1	0.373	0.584	0.051	0.024
		D+H		0.100	0.000	0.504	0.839
LATB	LATCH with complementary output (6)	H->Q	1	0.580	0.919	0.051	0.026
		H->QB	1	1.010	0.845	0.050	0.023
		D->Q	1	0.424	0.657	0.051	0.026
		D->QB	1	0.738	0.674	0.050	0.025
		D+H		0.000	0.000	0.496	0.755
LATBG	LATCH with complementary outputs and inverted gate signal (6)	G->Q	1	0.556	0.697	0.050	0.026
		G->QB	1	0.719	0.781	0.052	0.023
		D->Q	1	0.397	0.638	0.050	0.026
		D->QB	1	0.653	0.626	0.051	0.023
		D+G		0.100	0.173	0.190	0.893

ATL80 0.8 μ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Cell Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
LATBGQ	Quad LATBG with common gate signal (20)	G->Q0	2	0.807	0.739	0.051	0.027
		G->Q1	2	0.807	0.739	0.051	0.027
		G->Q2	2	0.807	0.739	0.051	0.027
		G->Q3	2	0.807	0.739	0.051	0.027
		G->Q0B	2	0.826	1.085	0.051	0.023
		G->Q1B	2	0.826	1.085	0.051	0.023
		G->Q2B	2	0.826	1.085	0.051	0.023
		G->Q3B	2	0.826	1.085	0.051	0.023
		D0->Q0	1	0.424	0.661	0.052	0.026
		D1->Q1	1	0.424	0.661	0.052	0.026
		D2->Q2	1	0.424	0.661	0.052	0.026
		D3->Q3	1	0.424	0.661	0.052	0.026
		D0->Q0B	1	0.726	0.704	0.051	0.022
		D1->Q1B	1	0.726	0.704	0.051	0.022
		D2->Q2B	1	0.723	0.704	0.052	0.022
		D3->Q3B	1	0.723	0.704	0.052	0.022
		D0+G		0.000	0.000	0.157	0.753
		D1+G		0.000	0.000	0.157	0.753
		D2+G		0.000	0.000	0.157	0.753
		D3+G		0.000	0.000	0.157	0.753
LATBH	LATCH with high drive complementary outputs (7)	H->Q	1	0.601	0.837	0.016	0.015
		H->QB	1	1.029	0.823	0.012	0.011
		D->Q	1	0.480	0.673	0.023	0.016
		D->QB	1	0.834	0.716	0.022	0.012
		D+H		0.000	0.000	0.585	0.951
LATIQ	Quad inverting LATCH (20)	D0->Q0	2	0.568	0.608	0.022	0.012
		D1->Q1	2	0.568	0.608	0.022	0.012
		D2->Q2	2	0.568	0.608	0.022	0.012
		D3->Q3	2	0.568	0.608	0.022	0.012
		H->Q0	1	0.927	1.043	0.023	0.013
		H->Q1	1	0.927	1.043	0.023	0.013
		H->Q2	1	0.927	1.043	0.023	0.013
		H->Q3	1	0.927	1.043	0.023	0.013
		D0+H		0.057	0.000	0.178	0.869
		D1+H		0.057	0.000	0.178	0.869
		D2+H		0.057	0.000	0.178	0.869
		D3+H		0.057	0.000	0.178	0.869
LATR	LATCH with reset (4)	D->Q	1	0.423	0.604	0.094	0.024
		H->Q	2	0.466	0.496	0.095	0.025
		R->Q	1	—	0.247	—	0.026
		D+H		0.000	0.000	0.693	1.012
LATS	LATCH with set (6)	D->Q	1	0.375	0.829	0.051	0.027
		H->Q	2	0.407	0.661	0.051	0.027
		S->Q	2	0.444	—	0.051	—
		D+H		0.000	0.000	0.506	1.082

ATL80 0.8 μ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Cell Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
LATSR	LATCH with set and reset (8)	D->Q	1	0.836	1.069	0.023	0.012
		H->Q	2	0.875	0.949	0.023	0.012
		R->Q	1	0.874	0.502	0.022	0.012
		S->Q	2	0.874	——	0.022	——
		D+H		0.000	0.000	0.601	1.428
MUX2	2:1 MUX (4)	I0->O	1	0.403	0.587	0.025	0.015
		I1->O	1	0.413	0.586	0.024	0.015
		S->O	2	0.421	0.427	0.023	0.015
MUX2H	2:1 MUX - high drive (5)	I0->O	1	0.528	0.719	0.012	0.010
		I1->O	1	0.524	0.722	0.012	0.009
		S->O	2	0.542	0.539	0.012	0.010
MUX2I	2:1 MUX with inverted output (3)	I0->O	2	0.158	0.244	0.053	0.029
		I1->O	2	0.164	0.245	0.051	0.029
		S->O	2	0.249	0.339	0.050	0.023
MUX2IH	2:1 MUX with inverted output - high drive (4)	I0->O	3	0.223	0.211	0.023	0.018
		I1->O	3	0.215	0.214	0.024	0.018
		S->O	2	0.325	0.372	0.023	0.012
MUX2N	2:1 MUX with active low enable (4)	I0->O	1	0.358	0.587	0.095	0.024
		I1->O	1	0.350	0.595	0.095	0.024
		S->O	2	0.340	0.412	0.095	0.024
		E->O	1	0.108	0.171	0.095	0.027
MUX2NQ	Quad 2:1 MUX with active low enable (18)	IA0->OA	1	0.405	0.599	0.093	0.024
		IA1->OA	1	0.406	0.599	0.093	0.024
		IB0->OB	1	0.405	0.599	0.093	0.024
		IB1->OB	1	0.406	0.599	0.093	0.024
		IC0->OC	1	0.405	0.599	0.093	0.024
		IC1->OC	1	0.406	0.599	0.093	0.024
		ID0->OD	1	0.405	0.599	0.093	0.024
		ID1->OD	1	0.406	0.599	0.093	0.024
		E->OA	1	0.478	0.311	0.095	0.016
		E->OB	1	0.478	0.311	0.095	0.016
		E->OC	1	0.478	0.311	0.095	0.016
		E->OD	1	0.478	0.311	0.095	0.016
		S->OA	1	0.386	0.599	0.095	0.024
		S->OB	1	0.386	0.599	0.095	0.024
		S->OC	1	0.386	0.599	0.095	0.024
		S->OD	1	0.386	0.599	0.095	0.024

ATL80 0.8 μ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

				Intercept		Slope	
Cell Name	Description (Site Count)	Path Timechk	Input Load	Rising Hold-R	Falling Hold-F	Rising Setup-R	Falling Setup-F
MUX2Q	Quad 2:1 MUX (14)	IA0->OA	1	0.352	0.577	0.051	0.024
		IA1->OA	1	0.352	0.577	0.051	0.024
		IB0->OB	1	0.352	0.577	0.051	0.024
		IB1->OB	1	0.352	0.577	0.051	0.024
		IC0->OC	1	0.352	0.577	0.051	0.024
		IC1->OC	1	0.352	0.577	0.051	0.024
		ID0->OD	1	0.352	0.577	0.051	0.024
		ID1->OD	1	0.352	0.577	0.051	0.024
		S->OA	1	0.856	1.023	0.050	0.024
		S->OB	1	0.856	1.023	0.050	0.024
		S->OC	1	0.856	1.023	0.050	0.024
		S->OD	1	0.856	1.023	0.050	0.024
MUX3I	3:1 MUX with inverted output (6)	I0->O	2	0.380	0.329	0.094	0.037
		I1->O	2	0.381	0.330	0.094	0.037
		I2->O	1	0.178	0.320	0.095	0.037
		S0->O	2	0.453	0.430	0.095	0.035
		S1->O	2	0.170	0.294	0.095	0.036
MUX3IH	3:1 MUX with inverted output - high drive (8)	I0->O	2	0.448	0.326	0.047	0.020
		I1->O	2	0.447	0.330	0.046	0.020
		I2->O	2	0.220	0.301	0.046	0.021
		S0->O	2	0.570	0.496	0.046	0.018
		S1->O	2	0.229	0.264	0.044	0.020
MUX4	4:1 MUX (9)	I0->O	1	0.529	0.971	0.051	0.029
		I1->O	1	0.541	0.977	0.050	0.029
		I2->O	1	0.523	0.976	0.051	0.029
		I3->O	1	0.524	0.980	0.051	0.029
		S0->O	3	0.527	0.714	0.051	0.029
		S1->O	2	0.496	0.377	0.051	0.028
MUX4X	4:1 MUX with transmission gate data inputs (10)	I0->O	2	0.403	0.618	0.052	0.025
		I1->O	2	0.413	0.607	0.051	0.025
		I2->O	2	0.403	0.620	0.052	0.025
		I3->O	2	0.414	0.616	0.051	0.025
		S0->O	3	0.539	0.806	0.052	0.025
		S1->O	2	0.308	0.369	0.052	0.025
MUX4XH	4:1 MUX with transmission gate data inputs - high drive (10)	I0->O	2	0.510	0.666	0.023	0.015
		I1->O	2	0.510	0.666	0.023	0.015
		I2->O	2	0.507	0.671	0.023	0.014
		I3->O	2	0.508	0.668	0.023	0.014
		S0->O	3	0.646	0.848	0.023	0.015
		S1->O	2	0.427	0.425	0.023	0.015

ATL80 0.8 μ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Cell Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
MUX5H	5:1 MUX - high drive (14)	I0->O	1	1.142	1.347	0.012	0.010
		I1->O	1	1.142	1.344	0.012	0.010
		I2->O	1	1.150	1.340	0.011	0.010
		I3->O	1	1.156	1.346	0.011	0.010
		I4->O	1	0.533	0.734	0.012	0.009
		S0->O	1	1.516	1.632	0.012	0.011
		S1->O	2	0.964	1.084	0.012	0.010
		S2->O	2	0.604	0.801	0.012	0.010
MUX8	8:1 MUX (18)	I0->O	1	1.074	1.377	0.014	0.014
		I1->O	1	1.075	1.377	0.014	0.014
		I2->O	1	1.074	1.376	0.014	0.014
		I3->O	1	1.074	1.376	0.014	0.014
		I4->O	1	1.074	1.375	0.014	0.014
		I5->O	1	1.074	1.376	0.014	0.014
		I6->O	1	1.074	1.378	0.014	0.014
		I7->O	1	1.074	1.377	0.014	0.014
		S0->O	1	1.603	1.822	0.014	0.014
		S1->O	1	1.078	1.051	0.013	0.014
		S2->O	1	0.763	0.669	0.014	0.012
MUX8N	8:1 MUX with active low enable (20)	I0->O	1	0.941	1.226	0.095	0.033
		I1->O	1	0.942	1.226	0.095	0.033
		I2->O	1	0.942	1.226	0.095	0.033
		I3->O	1	0.942	1.226	0.095	0.033
		I4->O	1	0.941	1.226	0.095	0.033
		I5->O	1	0.941	1.226	0.095	0.033
		I6->O	1	0.941	1.226	0.095	0.033
		I7->O	1	0.942	1.226	0.095	0.033
		S0->O	1	1.466	1.676	0.095	0.033
		S1->O	1	0.929	0.946	0.095	0.033
		S2->O	1	0.640	0.622	0.096	0.029
		E->O	1	0.557	0.171	0.094	0.022
MUX8XH	8:1 MUX with transmission gate data inputs - high drive (18)	I0->O	2	0.833	1.034	0.023	0.018
		I1->O	2	0.832	1.036	0.023	0.018
		I2->O	2	0.832	1.036	0.023	0.018
		I3->O	2	0.833	1.034	0.023	0.018
		I4->O	2	0.833	1.034	0.023	0.018
		I5->O	2	0.832	1.036	0.023	0.018
		I6->O	2	0.832	1.036	0.023	0.018
		I7->O	2	0.833	1.034	0.023	0.018
		S0->O	1	1.220	1.488	0.024	0.018
		S1->O	3	0.723	0.755	0.023	0.018
		S2->O	2	0.623	0.419	0.024	0.017

4

ATL80 0.8 μ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Cell Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
NAN2	2 input NAND (2)	A->O	1	0.130	0.214	0.052	0.036
		B->O	1	0.085	0.250	0.052	0.037
NAN2D	Dual 2 input NAND (3)	A0->O0	1	0.130	0.214	0.052	0.036
		B0->O0	1	0.085	0.250	0.052	0.037
		A1->O1	1	0.130	0.214	0.052	0.036
		B1->O1	1	0.085	0.250	0.052	0.037
NAN2H	2 input NAND - high drive (2)	A->O	2	0.176	0.203	0.023	0.019
		B->O	2	0.123	0.234	0.024	0.020
NAN3	3 input NAND (2)	A->O	1	0.192	0.284	0.052	0.047
		B->O	1	0.145	0.305	0.052	0.047
		C->O	1	0.090	0.308	0.052	0.049
NAN3H	3 input NAND - high drive (3)	A->O	2	0.239	0.288	0.023	0.024
		B->O	2	0.193	0.295	0.023	0.025
		C->O	2	0.133	0.297	0.024	0.026
NAN4	4 input NAND (3)	A->O	1	0.236	0.381	0.052	0.059
		B->O	1	0.205	0.386	0.052	0.059
		C->O	1	0.163	0.383	0.052	0.060
		D->O	1	0.102	0.370	0.053	0.060
NAN4H	4 input NAND - high drive (4)	A->O	2	0.217	0.372	0.023	0.030
		B->O	2	0.217	0.372	0.023	0.030
		C->O	2	0.224	0.391	0.024	0.030
		D->O	2	0.224	0.391	0.024	0.030
NAN5	5 input NAND (5)	A->O	1	0.427	0.704	0.051	0.023
		B->O	1	0.376	0.605	0.051	0.024
		C->O	1	0.370	0.711	0.051	0.024
		D->O	1	0.312	0.636	0.052	0.023
		E->O	1	0.293	0.714	0.052	0.023
NAN5H	5 input NAND - high drive (6)	A->O	1	0.509	0.775	0.023	0.014
		B->O	1	0.463	0.665	0.023	0.014
		C->O	1	0.466	0.780	0.022	0.014
		D->O	1	0.387	0.681	0.023	0.014
		E->O	1	0.386	0.774	0.023	0.014

ATL80 0.8 μ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Cell Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
NAN6	6 input NAND (6)	A->O	1	0.443	0.735	0.052	0.023
		B->O	1	0.425	0.704	0.051	0.023
		C->O	1	0.405	0.747	0.051	0.023
		D->O	1	0.370	0.711	0.051	0.024
		E->O	1	0.335	0.728	0.050	0.024
		F->O	1	0.292	0.714	0.052	0.023
NAN6H	6 input NAND - high drive (7)	A->O	1	0.542	0.781	0.022	0.014
		B->O	1	0.508	0.775	0.023	0.014
		C->O	1	0.489	0.796	0.023	0.014
		D->O	1	0.466	0.780	0.022	0.014
		E->O	1	0.407	0.791	0.023	0.014
		F->O	1	0.386	0.774	0.023	0.014
NAN8	8 input NAND (7)	A->O	1	0.596	0.914	0.022	0.014
		B->O	1	0.570	0.918	0.022	0.014
		C->O	1	0.548	0.930	0.023	0.013
		D->O	1	0.535	0.926	0.022	0.014
		E->O	1	0.509	0.919	0.022	0.014
		F->O	1	0.482	0.921	0.022	0.014
		G->O	1	0.436	0.886	0.023	0.014
		H->O	1	0.406	0.883	0.023	0.014
NAN8H	8 input NAND - high drive (7)	A->O	1	0.728	1.118	0.011	0.009
		B->O	1	0.711	1.121	0.010	0.009
		C->O	1	0.700	1.125	0.011	0.009
		D->O	1	0.669	1.131	0.010	0.009
		E->O	1	0.626	1.120	0.011	0.009
		F->O	1	0.599	1.123	0.011	0.009
		G->O	1	0.557	1.084	0.011	0.009
		H->O	1	0.540	1.087	0.011	0.009
NOR2	2 input NOR (2)	A->O	1	0.103	0.195	0.095	0.028
		B->O	1	0.121	0.157	0.094	0.028
NOR2D	Dual 2 input NOR (3)	A0->O0	1	0.103	0.195	0.095	0.028
		B0->O0	1	0.121	0.157	0.094	0.028
		A1->O1	1	0.103	0.195	0.095	0.028
		B1->O1	1	0.121	0.157	0.094	0.028
NOR2H	2 input NOR - high drive (2)	A->O	2	0.172	0.176	0.045	0.016
		B->O	2	0.170	0.126	0.045	0.016

ATL80 0.8 μ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Cell Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
NOR3	3 input NOR (2)	A->O	1	0.344	0.244	0.135	0.028
		B->O	1	0.328	0.224	0.136	0.028
		C->O	1	0.305	0.193	0.135	0.028
NOR3H	3 input NOR - high drive (3)	A->O	2	0.308	0.185	0.068	0.016
		B->O	2	0.281	0.176	0.067	0.016
		C->O	2	0.232	0.143	0.067	0.016
NOR4	4 input NOR (3)	A->O	1	0.558	0.252	0.177	0.027
		B->O	1	0.518	0.254	0.177	0.026
		C->O	1	0.448	0.235	0.177	0.026
		D->O	1	0.322	0.202	0.176	0.026
NOR4H	4 input NOR - high drive (4)	A->O	2	0.430	0.177	0.087	0.016
		B->O	2	0.430	0.177	0.087	0.016
		C->O	2	0.448	0.192	0.089	0.016
		D->O	2	0.448	0.192	0.089	0.016
NOR5	5 input NOR (5)	A->O	1	0.735	0.493	0.050	0.023
		B->O	1	0.698	0.484	0.052	0.023
		C->O	1	0.632	0.431	0.053	0.023
		D->O	1	0.484	0.412	0.051	0.022
		E->O	1	0.474	0.358	0.051	0.022
NOR8	8 input NOR (7)	A->O	1	1.028	0.517	0.050	0.023
		B->O	1	1.019	0.451	0.051	0.022
		C->O	1	0.987	0.516	0.051	0.023
		D->O	1	1.003	0.456	0.050	0.022
		E->O	1	0.882	0.493	0.053	0.023
		F->O	1	0.901	0.422	0.052	0.023
		G->O	1	0.781	0.451	0.051	0.023
		H->O	1	0.775	0.394	0.051	0.022

ATL80 0.8 μ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Cell Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
NOR16H	16 input NOR - high drive (13)	A->O	1	1.368	0.527	0.024	0.013
		B->O	1	1.327	0.528	0.024	0.013
		C->O	1	1.244	0.503	0.024	0.013
		D->O	1	1.137	0.482	0.024	0.012
		E->O	1	1.371	0.584	0.024	0.013
		F->O	1	1.342	0.574	0.024	0.013
		G->O	1	1.270	0.572	0.024	0.013
		H->O	1	1.146	0.519	0.024	0.013
		I->O	1	1.387	0.627	0.024	0.014
		J->O	1	1.347	0.621	0.024	0.014
		K->O	1	1.268	0.607	0.025	0.013
		L->O	1	1.159	0.565	0.024	0.014
		M->O	1	1.397	0.664	0.024	0.014
		N->O	1	1.357	0.646	0.024	0.014
		O->O	1	1.284	0.631	0.024	0.014
		P->O	1	1.155	0.596	0.024	0.014
OAI22	2 input OR into 2 input NAND (2)	A->O	1	0.212	0.226	0.094	0.037
		B->O	1	0.219	0.269	0.094	0.036
		C->O	1	0.100	0.242	0.051	0.038
OAI22H	2 Input OR into 3 input NAND - high drive (4)	A->O	2	0.359	0.214	0.044	0.021
		B->O	2	0.357	0.277	0.044	0.019
		C->O	2	0.155	0.207	0.023	0.019
OAI222	Two, 2 input ORs into 2 input NAND (2)	A->O	1	0.326	0.296	0.095	0.036
		B->O	1	0.325	0.337	0.095	0.036
		C->O	1	0.169	0.346	0.095	0.037
		D->O	1	0.170	0.294	0.095	0.037
OAI222H	Two, 2 input ORs into 2 input NAND - high drive (4)	A->O	2	0.374	0.284	0.045	0.019
		B->O	2	0.376	0.331	0.045	0.019
		C->O	2	0.226	0.330	0.045	0.020
		D->O	2	0.216	0.276	0.046	0.020

ATL80 0.8 μ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Cell Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
OAI22224	Four, 2 input ORs into 4 input NAND (6)	A->O	1	0.700	0.716	0.051	0.023
		B->O	1	0.699	0.759	0.052	0.023
		C->O	1	0.527	0.746	0.051	0.024
		D->O	1	0.517	0.694	0.052	0.024
		E->O	1	0.654	0.695	0.052	0.023
		F->O	1	0.672	0.735	0.052	0.024
		G->O	1	0.489	0.732	0.052	0.024
		H->O	1	0.481	0.672	0.053	0.023
OAI23	2 input OR into 3 input NAND (3)	A->O	1	0.153	0.368	0.095	0.049
		B->O	1	0.155	0.304	0.095	0.050
		C->O	1	0.236	0.317	0.050	0.048
		D->O	1	0.196	0.343	0.050	0.048
ORR2	2 input OR (2)	A->O	1	0.198	0.419	0.052	0.023
		B->O	1	0.147	0.401	0.051	0.024
ORR2H	2 input OR - high drive (3)	A->O	1	0.365	0.545	0.014	0.010
		B->O	1	0.308	0.540	0.015	0.010
ORR3	3 input OR (3)	A->O	1	0.339	0.715	0.023	0.016
		B->O	1	0.300	0.696	0.023	0.015
		C->O	1	0.264	0.625	0.023	0.016
ORR3H	3 input OR - high drive (4)	A->O	1	0.438	0.913	0.011	0.010
		B->O	1	0.417	0.889	0.011	0.010
		C->O	1	0.366	0.828	0.011	0.010
ORR4	4 input OR (3)	A->O	1	0.261	0.920	0.051	0.029
		B->O	1	0.256	0.890	0.050	0.029
		C->O	1	0.217	0.802	0.052	0.029
		D->O	1	0.195	0.683	0.051	0.029

ATL80 0.8 μ Cell Index (Typical Delays at Tj = 25°C; Vdd = 5.0 V; Input Rise and Fall Times = 1 ns; Process = Nominal)

Cell Name	Description (Site Count)	Path Timechk	Input Load	Intercept Rising Hold-R	Intercept Falling Hold-F	Slope Rising Setup-R	Slope Falling Setup-F
ORR4H	4 input OR - high drive (4)	A->O	1	0.403	1.139	0.016	0.014
		B->O	1	0.404	1.099	0.015	0.014
		C->O	1	0.378	1.025	0.015	0.014
		D->O	1	0.331	0.911	0.015	0.014
ORR5	5 input OR (5)	A->O	1	0.248	0.694	0.051	0.037
		B->O	1	0.262	0.452	0.051	0.035
		C->O	1	0.223	0.665	0.051	0.036
		D->O	1	0.216	0.437	0.050	0.036
		E->O	1	0.177	0.608	0.051	0.037
XNR2	2 input exclusive NOR (4)	A->O	2	0.172	0.232	0.095	0.036
		B->O	2	0.078	0.330	0.095	0.037
XNR2H	2 input exclusive NOR - high drive (4)	A->O	1	0.549	0.729	0.023	0.015
		B->O	2	0.457	0.453	0.022	0.014
XOR2	2 input exclusive OR (4)	A->O	2	0.382	0.351	0.094	0.034
		B->O	2	0.201	0.282	0.095	0.034
XOR2H	2 input exclusive OR - high drive (4)	A->O	1	0.487	0.522	0.024	0.015
		B->O	2	0.424	0.396	0.023	0.015

4

I/O Cells

Sample of buffers composed of modular I/O building blocks

Cell Name	Description	Path Timechk	Input Load	Intercept Rising	Intercept Falling	Slope Rising	Slope Falling
PBS2C	4 mA bidi CMOS buffer	AO->P	2.4	1.125	1.796	0.080	0.077
		E0->P	2.4	1.063	2.005	0.080	0.079
		P->AI0		0.756	0.766	0.005	0.005
PBS2T	4 mA bidi TTL buffer	AO->P	2.4	1.125	1.796	0.080	0.077
		E0->P	2.4	1.063	2.005	0.080	0.079
		P->AI0		0.703	0.809	0.005	0.005
PBS2TU3	4 mA bidi buffer with single enable TTL pullup	AO->P	2.4	1.125	1.796	0.080	0.077
		EO->P	2.4	1.063	2.005	0.081	0.079
		PI->AIO		0.670	0.859	0.005	0.006
PBS2TU31	4 mA bidi TTL buffer with pullup	AO->P	2.4	1.125	1.796	0.080	0.077
		E0->P	2.4	1.063	2.005	0.080	0.079
		P->AI0		0.706	0.832	0.005	0.006
PBS3T	6 mA bidi buffer with single enable TTL	AO->P	2.4	1.442	1.467	0.054	0.050
		EO->P	2.4	1.367	1.674	0.054	0.051
		P->AIO		0.791	0.819	0.005	0.005
PBS4T	8 mA bidi TTL buffer	AO->P	2.4	1.616	1.591	0.041	0.043
		E0->P	2.4	1.543	1.730	0.041	0.044
		P->AI0		0.703	0.809	0.005	0.005
PBS6T	12 mA bidi TTL buffer	AO->P	2.4	1.979	2.069	0.030	0.030
		E0->P	2.4	1.884	2.037	0.030	0.032
		P->AI0		0.703	0.809	0.005	0.005
PBS6TU3	12 mA bidi TTL buffer, with pullup	AO->P	2.4	1.979	2.069	0.030	0.030
		E0->P	2.4	1.884	2.037	0.030	0.032
		P->AI0		0.651	0.855	0.005	0.005
PIC	CMOS input buffer	P->AI0		0.756	0.766	0.005	0.005
PIT	TTL input buffer	P->AI0		0.703	0.809	0.005	0.005
PITS	TTL input buffer with Schmitt Trigger	P->AIO		1.598	2.220	0.012	0.012

I/O Cells

Cell Name	Description	Path Timechk	Input Load	Intercept Rising	Intercept Falling	Slope Rising	Slope Falling
PITU31	TTL input buffer with pullup	P->AI0		0.706	0.832	0.005	0.006
PK3	6 mA clock buffer	I->O		0.480	0.721	0.002	0.003
PO2	4 mA output buffer	AO->P	2.4	1.125	1.796	0.080	0.077
PO3	6 mA output buffer	AO->P	2.4	1.451	1.472	0.054	0.050
PO4	8 mA output buffer	AO->P	2.4	1.616	1.591	0.041	0.043
PO6	12 mA output buffer	AO->P	2.4	1.979	2.069	0.030	0.030
PTS2	4 mA tristate output buffer	AO->P EO->P	2.4 2.4	1.217 1.156	1.933 2.139	0.080 0.081	0.077 0.078
PTS4	8 mA tristate output buffer	AO->P EO->P	2.4 2.4	1.658 1.585	1.638 1.782	0.041 0.041	0.043 0.044
PTS6	12 mA tristate output buffer	AO->P E0->P	2.4 2.4	1.979 1.884	2.069 2.037	0.030 0.030	0.030 0.032
PX2CR	4 mA crystal oscillator buffer	PI->AI0 PI->PO		1.482 1.157	1.632 1.389	0.005 0.080	0.006 0.084

ATMEL

Section 5 PLD Application Notes

Product-Specific Application Briefs

Application Notes

Using the ATV750 with Abel™ and Cupl™

Typical applications for first and second generation PLDs include address decoding and counting. Here is an example using the ATV750, a third generation PLD, to implement a more complex counter. The following pages show example input listings for Abel™ and Cupl™.

The first listing is for Abel™ 3.0. The second listing is for Cupl™ 2.15b.

This design uses all twenty flip-flops of the ATV750 to build a 20-bit synchronous /asynchronous counter. With COUNT high and COUNT10, PRESET, RESET, Q1SEL, and OE low, a 1-MHz signal on the CLK pin will produce roughly a 1-Hz signal on pin 23.

The unique architecture of the macrocell gives the ATV750 its versatility and also increases gate utilization. Each of the twenty registers has its Q and $\overline{Q}$ feeding back to the array.

The output registers (Q0's) can be addressed directly (by the pin names in Abel™; in Cupl™ define PINNODES 35 through 44). However, to access the buried registers (Q1's), the corresponding nodes have to be named (nodes 26 through 35 in Abel™, PINNODEs 25 through 34 in Cupl™). They are called B14, B15,...B24 to show the correspondence with their Q0 counterparts. Any valid identifier can be used as a node name.

The ATV750 provides a global synchronous Preset which is accessible through the node definition or by extension. Each of the twenty flip-flops has its own clock, reset and sum term (not like a second generation PLD that allows only one clock and one reset for all registers). Use the '.CK' extension in conjunction with the named registers to define the equations for the clock inputs for all the registers. Use the '.RE' command following the named registers to define the reset terms.

The ATV750 gives the user a total of eight choices to configure each output. The operators ':=' and '=' inform Abel™ the output is registered or combinatorial, respectively. In Cupl™, use of the .Q extension defines a registered output. Use the '!' operator to define an active low output; active high output is assumed by default. Another convenient method is to use the 'ISTYPE' statement (Abel™ only) to define the outputs as high/low and registered/combinatorial. (Note: the ATV22V10 is defined the same way.)

The ATV750 has an advanced feature that lets the user combine or separate the sum terms in each macrocell. By default, the terms are combined. In Cupl™ and Abel™, using the buried register automatically splits the sum terms.

Sets (Abel™) or *fields* (Cupl™) are often defined for ease of referencing a group of signals or constants. In this particular example, *OUTS* is a collection of outputs used in the OE definition. *Out* is a reserved word in Cupl, and *Ouch* was used instead.

When PRESET and COUNT are asserted and the Clk pin goes high, all registers, even the asynchronously clocked ones, will go to the 'one' state. This is because as each flip-flop goes high, it forces the clock of the next flip-flop high, rippling the preset condition throughout the entire bank. To reset the output registers and the buried registers, simply have RESET high and vary !Q1SEL accordingly.

The O registers pair with the corresponding B registers to form ten two-bit synchronous counters. These are clocked by the preceding pair's output, thus forming a 20-bit counter. The last product term for the 'O' logic changes this device into a ten-bit counter with the output register mimicking the B registers. This provides observability, and a handy test mode. Test vectors take full

High Density UV Erasable Programmable Logic Device

Application Brief

5

advantage of Abel™'s and Cupl™'s ability to simulate the device before programming. This feature can save hours of testing and enable the user to make the necessary changes in seconds without ever leaving his or her PC. Therefore, it is highly recommended to take the time to write a comprehensive set of test vectors; this will reduce the time spent on the test bench.

ABEL™ and CUPL™ may be trademarks of others.

Abel™ Example

```
module EX3
title '20 Bit Counter for Atmel's ATV750
EX375  device 'P750';
Clk                       pin   1;
COUNT,COUNT10,PRESET      pin   2, 3, 4;
RESET, Q1SEL , OE         pin   5, 11, 13;
O14,O15,O16,O17,O18       pin   14,15,16,17,18;
O19,O20,O21,O22,O23       pin   19,20,21,22,23;
B14,B15,B16,B17,B18       node  26,27,28,29,30;
B19,B20,B21,B22,B23       node  31,32,33,34,35;
"       Nodes       Description
"       26..35      Q1 for pins 14 to 23
"Sets
OUTS=[O23,O22,O21,O20,O19,O18,O17,O16,O15,O14];
H,L,Z,C,X       =     1,0,.Z.,.C.,.X.;
Equations
O14.RE          =     O14.Q & RESET & !Q1SEL;
B14.RE          =     B14 & RESET & Q1SEL;
O15.RE          =     O15.Q & RESET & !Q1SEL;
B15.RE          =     B15 & RESET & Q1SEL;
O16.RE          =     O16.Q & RESET & !Q1SEL;
B16.RE          =     B16 & RESET & Q1SEL;
O17.RE          =     O17.Q & RESET & !Q1SEL;
B17.RE          =     B17 & RESET & Q1SEL;
O18.RE          =     O18.Q & RESET & !Q1SEL;
B18.RE          =     B18 & RESET & Q1SEL;
O19.RE          =     O19.Q & RESET & !Q1SEL;
B19.RE          =     B19 & RESET & Q1SEL;
O20.RE          =     O20.Q & RESET & !Q1SEL;
B20.RE          =     B20 & RESET & Q1SEL;
O21.RE          =     O21.Q & RESET & !Q1SEL;
B21.RE          =     B21 & RESET & Q1SEL;
O22.RE          =     O22.Q & RESET & !Q1SEL;
B22.RE          =     B22 & RESET & Q1SEL;
O23.RE          =     O23.Q & RESET & !Q1SEL;
B23.RE          =     B23 & RESET & Q1SEL;
O14.C           =     Clk & COUNT;
B14.CK          =     Clk & COUNT; "Synchronous
O15.CK          =     O14.Q & COUNT;
B15.CK          =     O14.Q & COUNT; "Asynchronous
O16.CK          =     O15.Q & COUNT;
B16.CK          =     O15.Q & COUNT;
O17.CK          =     O16.Q & COUNT;
B17.CK          =     O16.Q & COUNT;
O18.CK          =     O17.Q & COUNT;
B18.CK          =     O17.Q & COUNT;
O19.CK          =     O18.Q & COUNT;
B19.CK          =     O18.Q & COUNT;
O20.CK          =     O19.Q & COUNT;
B20.CK          =     O19.Q & COUNT;
O21.CK          =     O20.Q & COUNT;
B21.CK          =     O20.Q & COUNT;
O22.CK          =     O21.Q & COUNT;
B22.CK          =     O21.Q & COUNT;
O23.CK          =     O22.Q & COUNT;
B23.CK          =     O22.Q & COUNT;
B14             :=    !B14;
!O14            :=    !O14.Q &  B14 & !COUNT10
                #     O14.Q & !B14 & !COUNT10
                #     !B14 & COUNT10;
B15             :=    !B15;
!O15            :=    !O15.Q & B15 & !COUNT10
                #     O15.Q & !B15 & !COUNT10
                #     !B15 & COUNT10;
B16             :=    !B16;
!O16            :=    !O16.Q & B16 & !COUNT10
                #     O16.Q & !B16 & !COUNT10
                #     !B16 & COUNT10;
B17             :=    !B17;
!O17            :=    !O17.Q & B17 & !COUNT10
                #     O17.Q & !B17 & !COUNT10
                #     !B17 & COUNT10;
B18             :=    !B18;
!O18            :=    !O18.Q & B18 & !COUNT10
                #     O18.Q & !B18 & !COUNT10
                #     !B18 & COUNT10;
B19             :=    !B19;
!O19            :=    !O19.Q &  B19 & !COUNT10
                #     O19.Q & !B19 & !COUNT10
                #     !B19 & COUNT10;
B20             :=    !B20;
!O20            :=    !O20.Q & B20 & !COUNT10
                #     O20.Q & !B20 & !COUNT10
                #     !B20 & COUNT10;
B21             :=    !B21;
!O21            :=    !O21.Q & B21 & !COUNT10
                #     O21.Q & !B21 & !COUNT10
                #     !B21 & COUNT10;
B22             :=    !B22;
!O22            :=    !O22.Q & B22 & !COUNT10
                #     O22.Q & !B22 & !COUNT10
                #     !B22 & COUNT10;
B23             :=    !B23;
!O23            :=    !O23.Q & B23 & !COUNT10
                #     O23.Q & !B23 & !COUNT10
                #     !B23 & COUNT10;
O23.PR          =     PRESET;
ENABLE OUTS     =     !OE;
End
```

Cupl™ Example

```
Name          EX75;
Company       Atmel;
Device        V750;
/*********************************************/
/** Allowable Target Device Types :V750    */
/*********************************************/
/**Inputs     **/
PIN [1..4]    =   [Clk,COUNT,COUNT10,PRESET];
PIN [5,11,13]=   RESET,Q1SEL,OE;
/**Outputs**/
PIN [14..23] =   [Q14..Q23]; /* Pin Outputs*/
/*The easiest way to access the buried nodes
in CUPL*/
/*Refer to the I/O pins by their pin names,
and the */
/* Q0 and Q1 outputs by their pinnode names*/
PINNODE [25..34] =   [B14..B23]; /* Q1 nodes*/
field BEES   = [B23..B14]; /*Q1 field*/
field Ohi    = [O23..O19]; /*output hi field*/
field Olo    = [O18..O14]; /*output low field*/
field Ouch   = [O23..O14]; /*outputfield*/
/** Logic Equations **/
/*  The Asynch. Reset terms use the .AR
extension*/
Q14.AR        =   Q14 &  RESET & !Q1SEL;
B14.AR        =   B14 &  RESET & Q1SEL;
Q15.AR        =   Q15 &  RESET & !Q1SEL;
B15.AR        =   B15 &  RESET & Q1SEL;
Q16.AR        =   Q16 &  RESET & !Q1SEL;
B16.AR        =   B16 &  RESET & Q1SEL;
Q17.AR        =   Q17 &  RESET & !Q1SEL;
B17.AR        =   B17 &  RESET & Q1SEL;
Q18.AR        =   Q18 &  RESET & !Q1SEL;
B18.AR        =   B18 &  RESET & Q1SEL;
Q19.AR        =   Q19 &  RESET & !Q1SEL;
B19.AR        =   B19 &  RESET & Q1SEL;
Q20.AR        =   Q20 &  RESET & !Q1SEL;
B20.AR        =   B20 &  RESET & Q1SEL;
Q21.AR        =   Q21 &  RESET & !Q1SEL;
B21.AR        =   B21 &  RESET & Q1SEL;
Q22.AR        =   Q22 &  RESET & !Q1SEL;
B22.AR        =   B22 &  RESET & Q1SEL;
Q23.AR        =   Q23 &  RESET & !Q1SEL;
B23.AR        =   B23 &  RESET & Q1SEL;
/* The Clock lines are accessed with the .CK
extension*/
Q14.CK        =   Clk & COUNT;
B14.CK        =   Clk & COUNT; /* Synchronous */
Q15.CK        =   Q14 & COUNT;
B15.CK        =   Q14 & COUNT; /* Asynchronous*/
Q16.CK        =   Q15 & COUNT;
B16.CK        =   Q15 & COUNT;
Q17.CK        =   Q16 & COUNT;
B17.CK        =   Q16 & COUNT;
Q18.CK        =   Q17 & COUNT;
B18.CK        =   Q17 & COUNT;
Q19.CK        =   Q18 & COUNT;
B19.CK        =   Q18 & COUNT;
Q20.CK        =   Q19 & COUNT;
B20.CK        =   Q19 & COUNT;
Q21.CK        =   Q20 & COUNT;
B21.CK        =   Q20 & COUNT;
Q22.CK        =   Q21 & COUNT;
B22.CK        =   Q21 & COUNT;
Q23.CK        =   Q22 & COUNT;
B23.CK        =   Q22 & COUNT;
B14.D         =   !B14;
Q14.D         =   !Q14 &  B14 & !COUNT10
              #   Q14 & !B14 & !COUNT10
              #   !B14 & COUNT10;
/* Equations for both the B and the Q
automatically*/
/*tells CUPL to set the 'split' SUM term's
architecture bit.*/
B15.D         =   !B15;
Q15.D         =   !Q15 & B15 & !COUNT10
              #   Q15 & !B15 & !COUNT10
              #   !B15 & COUNT10;
B16.D         =   !B16;
Q16.D         =   !Q16 & B16 & !COUNT10
              #   Q16 & !B16 & !COUNT10
              #   !B16 & COUNT10;
B17.D         =   !B17;
Q17.D         =   !Q17 & B17 & !COUNT10
              #   Q17 & !B17 & !COUNT10
              #   !B17 & COUNT10;
B18.D         =   !B18;
Q18.D         =   !Q18 & B18 & !COUNT10
              #   Q18 & !B18 & !COUNT10
              #   !B18 & COUNT10;
B19.D         =   !B19;
Q19.D         =   !Q19 & B19 & !COUNT10
              #   Q19 & !B19 & !COUNT10
              #   !B19 & COUNT10;
B20.D         =   !B20;
Q20.D         =   !Q20 & B20 & !COUNT10
              #   Q20 & !B20 & !COUNT10
              #   !B20 & COUNT10;
B21.D         =   !B21;
Q21.D         =   !Q21 & B21 & !COUNT10
              #   Q21 & !B21 & !COUNT10
              #   !B21 & COUNT10;
B22.D         =   !B22;
Q22.D         =   !Q22 &  B22 & !COUNT10
              #   Q22 & !B22 & !COUNT10
              #   !B22 & COUNT10;
B23.D         =   !B23;
Q23.D         =   !Q23 & B23 & !COUNT10
              #   Q23 & !B23 & !COUNT10
              #   !B23 & COUNT10;
/*Only one synch preset equation is required*/
O23.SP        =   PRESET;
/*Use the .OE extension for the OE product
term*/
O14.oe        =   !OE ;    O19.oe=!OE ;
O15.oe        =   !OE ;    O20.oe=!OE ;
O16.oe        =   !OE ;    O21.oe=!OE ;
O17.oe        =   !OE ;    O22.oe=!OE ;
O18.oe        =   !OE ;    O23.oe=!OE ;
```

High Density UV Erasable Programmable Logic Device

Application Brief

Using the ATV2500 with Abel™ and Cupl™

The following two examples show example headers to use when designing the ATV2500 with Abel™ or Cupl™.

For Abel™, the node numbers shown may be assigned any legal Abel™ label. The fuse numbers for combining the product terms are included.

For Cupl™, the pinnodes shown may be assigned any legal Cupl™ label. Combining the product terms is handled automatically.

Abel™ Example

```
module NODE2500
title 'Addressing 48 Registers in V2500
NODE2500 device 'P2500';
      @message 'for Abel on a PC/Clone';
"Inputs
I1,I2,I3                                  pin    1,2,3;
I17,I18,I19,I20,I21,I22,I23     pin    17,18,19,20,21,22,23;
I37,I38,I39,I40                  pin    37,38,39,40;
"I/Os
O4,O5,O6,O7,O8,O9                pin    4,5,6,7,8,9;
O11,O12,O13,O14,O15,O16          pin    11,12,13,14,15,16;
O24,O25,O26,O27,O28,O29          pin    24,25,26,27,28,29;
O31,O32,O33,O34,O35,O36          pin    31,32,33,34,35,36;

"Q2 Registers
"Node Name         Node      Number          Pin Associated With:
B4,B5,B6           node      41,42,43;       " pin 4 to pin 6
B7,B8,B9           node      44,45,46;       " pin 7 to pin 9
B11,B12,B13        node      47,48,49;       " pin 11 to pin 13
B14,B15,B16        node      50,51,52;       " pin 14 to pin 16
B24,B25,B26        node      53,54,55;       " pin 24 to pin 26
B27,B28,B29        node      56,57,58;       " pin 27 to pin 29
B31,B32,B33        node      59,60,61;       " pin 31 to pin 33
B34,B35,B36        node      62,63,64;       " pin 34 to pin 36
"Q1 Registers
"Node Name         Node      Number          Pin Associated With:
Q4,Q5,Q6           node      217,218,219;    " pin 4 to pin 6
Q7,Q8,Q9           node      220,221,222;    " pin 7 to pin 9
Q11,Q12,Q13        node      223,224,225;    " pin 11 to pin 13
Q14,Q15,Q16        node      226,227,228;    " pin 14 to pin 16
Q24,Q25,Q26        node      229,230,231;    " pin 24 to pin 26
Q27,Q28,Q29        node      232,233,234;    " pin 27 to pin 29
Q31,Q32,Q33        node      235,236,237;    " pin 31 to pin 33
Q34,Q35,Q36        node      238,239,240;    " pin 34 to pin 36
```

5

Cupl™ Example

```
Name        NODE2500
Partno      00;
Date        11/21/88;
Revision    00;
Designer    J. Yu
Company     Atmel;
Assembly    None;
Location    None;
Device      V2500;
/*******************************************/
/** Allowable Target Device Types : V2500 **/
/*******************************************/
/**  Inputs  **/
/* This is a handy way to name a set of pins*/
PIN [1..3]       = [I1..I3];
PIN [17..23]     = [I17..I23];
PIN [37..40]     = [I37..I40];
/** I/Os           **
PIN [4..9]       = [O4..O9];
PIN [11..16]     = [O11..O16];
PIN [24..29]     = [O24..O29];
PIN [31..36]     = [O31..O36];
/** Declarations and Intermediate Variable
Definitions */
/* Q2 nodes                        Pin assoc. with:*/
PINNODE [41..46] = [B4..B9];      /*PIN 4 to 9*/
PINNODE [47..52] = [B11..B16];    /*PIN 11 to 16*/
PINNODE [53..58] = [B24..B29];    /*PIN 24 to 29*/
PINNODE [59..64] = [B31..B36];    /*PIN 31 to 36*/
/* Q1 nodes                        Pin assoc. with:*/
PINNODE [65..70] = [Q4..Q9];      /*PIN 4 to 9*/
PINNODE [71..76] = [Q11..Q16];    /*PIN 11 to 16*/
PINNODE [77..82] = [Q24..Q29];    /*PIN 24 to 29*/
PINNODE [83..88] = [Q31..Q36];    /*PIN 31 to 36*/
etc.
```

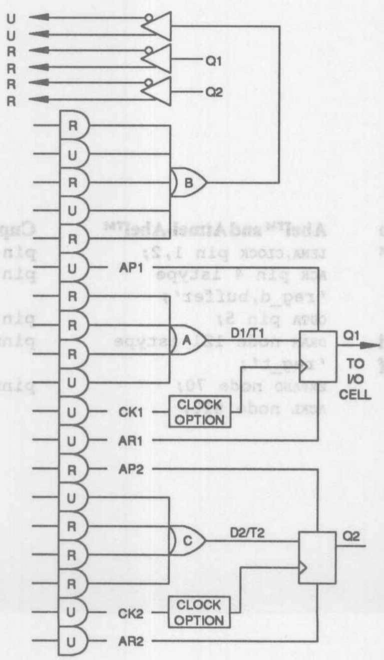
Abel™ and Cupl™ may be trademarks of others.

Using the ATV5000

The ATV5000 logic cells retain the ATV2500's basic features with two registers, three sum terms and Q1, Q2, and pin feedback options. In addition, these following features make the ATV5000 even more versatile:

1. Input latches are added to every I/O pin. Each latch can be enabled or disabled independently.
2. The ATV5000 can be configured to feedback a combinatorial sum term and send Q1 to the output. Feeding the combinatorial sum term internally makes implementing logic equations with large number of product terms possible without sacrificing an I/O pin.
3. Dedicated clock pins and individual clock product terms create multiple clock options for each flip-flop.
4. The addition of the buried logic cells handles more logic than ever.
5. Each flip-flop can be configured as D- or T-type flip-flop.
6. Asynchronous preset, asynchronous reset and programmable output polarity allow registered outputs using fewer product terms.

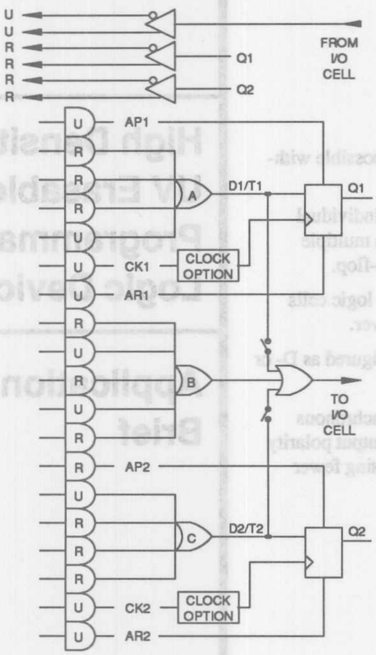
Logic Cell with Buried Sum Term and Register to I/O Cell

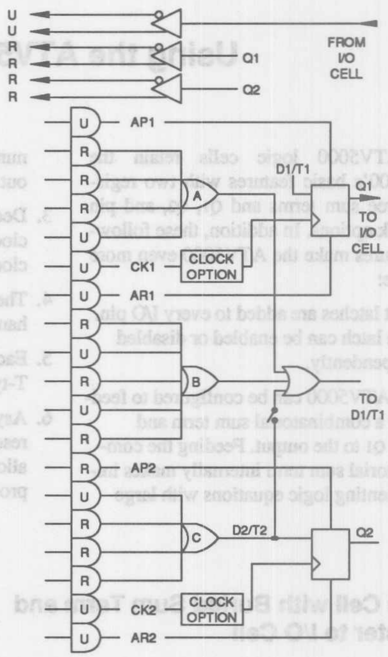


High Density UV Erasable Programmable Logic Device

Application Brief

5





The following are sections of Abel™ and Cupl™ source files to illustrate how each of these features is described in the Abel™ and Cupl™ high level description languages.

Pin and Node Assignments

All the buried registers used in the design need to be assigned node numbers. The following tables show the complete set of node numbers by quadrant.

Abel™ and Atmel-Abel™

```
LENA,CLOCK pin 1,2;
ACK pin 4 istype
'reg_d,buffer';
OUTA pin 5;
DRAM node 121 istype
'reg_t';
EXPAND node 70;
ACKL node 813;
```

Cupl™

```
pin 1,2 = LENA,CLOCK;
pin 4 = ACK;

pin 5 = OUTA;
pinnode 105 = DRAM;

pinnode 208 = EXPAND;
```

Quadrant I

Pin	Input Latch	Sum Term A	Sum Term B	Sum Term C
4	813	761	69	121
5	814	762	70	122
6	815	763	71	123
7	816	764	72	124
8	817	765	73	125
9	818	766	74	126
10	819	767	75	127
11	820	768	76	128
12	821	769	77	129
13	822	770	78	130
14	823	771	79	131
15	824	772	80	132
17	825	773	81	133

6 Buried Logic Cells
B23 - B18 node 173 - 178
LCK1 pin 1; Quadrant Latch clock
RCK1 pin 2; Quadrant Synchronous Register Clock

Quadrant II

Pin	Input Latch	Sum Term A	Sum Term B	Sum Term C
18	826	774	82	134
19	827	775	83	135
21	828	776	84	136
22	829	777	85	137
23	830	778	86	138
24	831	779	87	139
25	832	780	88	140
26	833	781	89	141
27	834	782	90	142
28	835	783	91	143
29	836	784	92	144
30	837	785	93	145
31	838	786	94	146

6 Buried Logic Cells
B17 - B12 node 179 - 184
LCK2 pin 34; Quadrant Latch Clock
RCK2 pin 32; Quadrant Synchronous Register Clock

Quadrant III

Pin	Input Latch	Sum Term A	Sum Term B	Sum Term C
38	839	787	95	147
39	840	788	96	148
40	841	789	97	149
41	842	790	98	150
42	843	791	99	151
43	844	792	100	152
44	845	793	101	153
45	846	794	102	154
46	847	795	103	155
47	848	796	104	156
48	849	797	105	157
49	850	798	106	158
51	851	799	107	159

6 Buried Logic Cells
B11 - B6 node 185 - 190
LCK3 pin 35; Quadrant Latch Clock
RCK3 pin 36; Quadrant Synchronous Register Clock

Quadrant IV

Pin	Input Latch	Sum Term A	Sum Term B	Sum Term C
52	852	800	108	160
53	853	801	109	161
55	854	802	110	162
56	855	803	111	163
57	856	804	112	164
58	857	805	113	165
59	858	806	114	166
60	859	807	115	167
61	860	808	116	168
62	861	809	117	169
63	862	810	118	170
64	863	811	119	171
65	864	812	120	172

6 Buried Logic Cells
B5 - B0 node 191 - 196
LCK4 pin 68; Quadrant Latch Clock
RCK4 pin 66; Quadrant Synchronous Register Clock

ATV5000 Cupl™ Node Numbers

Quadrant I

Pin	Sum Term A	Sum Term B	Sum Term C
4	157	209	105
5	156	208	104
6	155	207	103
7	154	206	102
8	153	205	101
9	152	204	100
10	151	203	99
11	150	202	98
12	149	201	97
13	148	200	96
14	147	199	95
15	146	198	94
17	145	197	93

6 Buried Logic Cells
B23 - B18 node 74 - 69
LCK1 pin 1; Quadrant Latch Clock
RCK1 pin 2; Quadrant Synchronous Register Clock

Quadrant II

Pin	Sum Term A	Sum Term B	Sum Term C
18	158	210	106
19	159	211	107
21	160	212	108
22	161	213	109
23	162	214	110
24	163	215	111
25	164	216	112
26	165	217	113
27	166	218	114
28	167	219	115
29	168	220	116
30	169	221	117
31	170	222	118

6 Buried Logic Cells
B17 - B12 node 75 - 80
LCK2 pin 34; Quadrant Latch Clock
RCK2 pin 32; Quadrant Synchronous Register Clock

Quadrant III

Pin	Sum Term A	Sum Term B	Sum Term C
38	183	235	131
39	182	234	130
40	181	233	129
41	180	232	128
42	179	231	127
43	178	230	126
44	177	229	125
45	176	228	124
46	175	227	123
47	174	226	122
48	173	225	121
49	172	224	120
51	171	223	119

6 Buried Logic Cells
B11 - B6 node 86 - 81
LCK3 pin 35; Quadrant Latch Clock
RCK3 pin 36; Quadrant Synchronous Register Clock

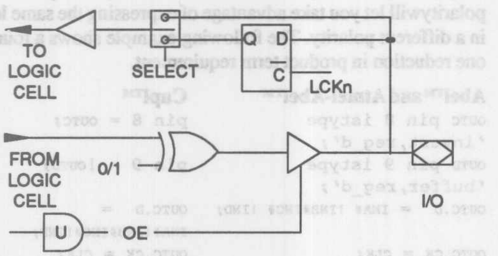
Quadrant IV

Pin	Sum Term A	Sum Term B	Sum Term C
52	184	236	132
53	185	237	133
55	186	238	134
56	187	239	135
57	188	240	136
58	189	241	137
59	190	242	138
60	191	243	139
61	192	244	140
62	193	245	141
63	194	246	142
64	195	247	143
65	196	248	144

6 Buried Logic Cells
B5 - B0 node 87 - 92
LCK4 pin 68; Quadrant Latch Clock
RCK4 pin 66; Quadrant Synchronous Register Clock

Input Latch

Each of the 52 I/Os has an input latch that can be enabled and disabled individually. When the latch is enabled and latch clock is high, the pin value is latched. When the quadrant latch clock is low, the latch becomes transparent. When the latch is disabled, an ATV5000 I/O acts like those of the ATV750 and ATV2500 I/Os. The pin input is fed directly to the array (except when sum term B is being used as a buried feedback).



The .D and the .LE (latch enable) equations are required to enable the input latch in Abel™ AHDL. The only allowed input to the latch is the IO pin with which it is associated. The only allowed .LE input is the quadrant latch clock (pin 1,34, 35, or 68). Notice that Cupl™ does not have node numbers for the input latches. Any pin name used in a feedback with the dot extension IOL tells Cupl™ that particular pin should be a latched pin.

Abel™ and Atmel-Abel™	Cupl™
`ACKL.D = ACK;`	
`ACKL.LE = LENA;`	
`OUTA := ACKL;`	`OUTA.D = ACK.IOL;`

Internal Combinatorial Feedback for an I/O Cell

To implement the combinatorial feedback of the B sum term, first define the node name with the corresponding node number. This node will take a five-product term equation. Regular syntax describing a combinatorial equation will describe the B sum term.

Note: This B sum term node number is defined only when this feature is needed. When this feature is used, the output is from Q1 through an inverter/buffer. *The I/O pin becomes a output-only pin. It cannot be used as an input or as an input/output.*

Abel™ and Atmel-Abel™	Cupl™
`OUTA pin 5;`	`pin 5 = OUTA;`
`EXPAND node 70;`	`pinnode 208 = EXPAND;`
`OUTA.D = INA&INB# !INC&!IND;`	`OUTA.D = INA&INB#!INC&!IND;`
`EXPAND = INA & !INC # IND;`	`EXPAND = INA & !INC # IND;`

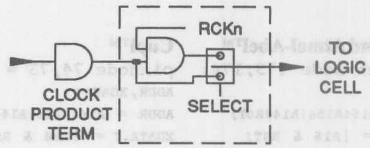
(OUTA and EXPAND are in the same I/O logic cell. OUTA is the Q1 output to pin after the inverter/buffer and EXPAND is the combinatorial sum term feedback. The feedbacks from this logic cell are Q1 before the inverter/buffer, Q2, and the B sum term)

Clocking Options

There are different methods of clocking the registers in the ATV5000. The clock options can best described as either the AND function of (quadrant clock & clock product term) or purely the function of the clock product term (like ATV750 and ATV2500).

Synchronous Operation

The quadrant clock is the only clocking element in this mode of operation. The clock product term must be defined to be equal to 1. In Abel AHDL, .CK defines the clock product term and .CE is the corresponding quadrant clock.



In Cupl™, .CE has a different implication. The keyword .CE means the user wants the AND function (quadrant clock & clock product term) for the clock.

Abel™ and Atmel-Abel™	Cupl™
`OUTA.CK = 1;`	
`OUTA.CE = CLOCK; "CLOCK is pin 2`	`OUTA.CE = 'B'1;`

Gated Synchronous Operation

This clock option still uses the fast quadrant register clock pin, but now it has a gating element. The clock product term enables or disables the clock going to the register.

Abel™ and Atmel-Abel™	Cupl™
`OUTA.CK = INX & INY & !INZ;`	
`OUTA.CE = CLOCK; "CLOCK is pin 2`	`OUTA.CE = INX&INY&!INZ;`

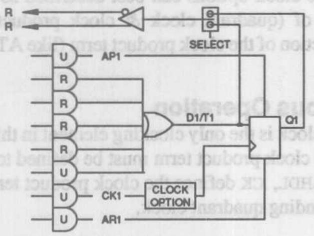
Asynchronous Operation

The quadrant clock has no effect on the register in this mode of clock option. The register is clocked by the clock product term like the ATV750 and ATV2500 registers.

Abel™ and Atmel-Abel™	Cupl™
`OUTA.CK = INA & !INB & INC;`	`OUTA.CK = INA & !INB & INC;`

Buried Logic Cells

There are six buried logic cells in each quadrant. Every buried logic cell can be configured as a buried register or as a combinatorial feedback to the quadrant array. In the buried register mode, it's used the same way as Q1 and Q2. In the combinatorial feedback mode, it's used the same way as the B sum term feedback.



Abel™ and Atmel-Abel™	Cupl™
`ADDR,XDATA node 173,174;`	`pinnode 74,73 = ADDR,XDATA;`
`ADDR = A16&A15&!A14#RST;`	`ADDR = A16&A15&!A14#RST;`
`XDATA.T = !A16 & RST;`	`XDATA.T = !A16 & RST;`
`XDATA.CK = INA & INC;`	`XDATA.CK = INA & INC;`
`XDATA.AR = INB;`	`XDATA.AR = INB;`

D-Type and T-Type Registers

All the registers in the ATV5000 can be configured as D- or T-type. The ISTYPE statement after the PIN or NODE definition in Abel™ AHDL tells Abel™ which type of register is needed.

Abel™ and Atmel-Abel™	Cupl™
`ABC node 123 istype 'reg_t';`	`pinnode 103 = ABC;`
`XYZ pin 7 istype 'reg_d';`	`pin 7 = XYZ;`
`ABC.T = INA & INB # INC;`	`ABC.T = INA & INB # INC;`
`XYZ.D = !INA # INC;`	`XYZ.D = !INA # INC;`

In both languages, the .T extension is reserved for T flip-flop and .D extension for D flip-flop.

Asynchronous Preset

One of the advantanges of having a programmable output polarity I/O is that the same combinatorial output often take fewer product terms to implement using negative logic rather than positive logic or vice versa.

The same can be done for a registered output provided the power-up state is not crucial (all flip-flops are reset upon power up). With an asynchronous preset and an asynchronous reset, simply swap the two product terms and invert your equation and polaritywill let you take advantage of expressing the same logic in a different polarity. The following example shows a four-to-one reduction in product term requirement.

Abel™ and Atmel-Abel™	Cupl™
`OUTC pin 8 istype 'invert,reg_d';`	`pin 8 = OUTC;`
`OUTD pin 9 istype 'buffer,reg_d';`	`pin 9 = !OUTD;`
`OUTC.D = INA# !INB#INC# !IND;`	`OUTC.D = INA#!INB#INC#!IND;`
`OUTC.CK = CLK;`	`OUTC.CK = CLK;`
`OUTC.AR = !INA & RST;`	`OUTC.AR = !INA & RST;`
`OUTC.AP = INB & PRESET;`	`OUTC.AP = INB & PRESET;`
`OUTD.D = !INA&INB&!INC&IND;`	`OUTD.D = !INA&INB&!INC&IND;`
`OUTD.CK = CLK;`	`OUTD.CK = CLK;`
`OUTD.AR = INB & PRESET;`	`OUTD.AR = INB & PRESET;`
`OUTD.AP = !INA & RST;`	`OUTD.AP = !INA & RST;`

Abel™ and Cupl™ may be trademarks of others.

Example Abel™ Description File

```
module V5000;
title    'Demo ATV5000 features with Atmel-abel (IBM386 or compatible)
          Atmel Corporation PLD    Joe Yu          April 28, 1991'
V5K device 'P5000';
"  The IOs, registers, latches, inputs, and combinatorial sum terms
"  feedbacks are named with the following prefixes for clarity:
"  Prefix
"  I    -  Quadrant Clocks, Latch Enables.
"  IO   -  IO pins
"  IL     -  Input Latches
"  BLC  - Buried Logic Cells
"  STF    -  Sum Term Feedbacks
"  Valid Abel AHDL identifiers can be used in place of them.
declarations
I1 pin 1;     " Quadrant 1  Latch Enable/Input
I2 pin 2;  " Quadrant 1  Synchronous Register Clock/Input
I32 pin 32;    " Quadrant 2  Synchronous Register Clock/Input
I34 pin 34;    " Quadrant 2  Latch Enable/Input
I35 pin 35;    " Quadrant 3  Latch Enable/Input
I36 pin 36;    " Quadrant 3  Synchronous Register Clock/Input
I66 pin 66;  " Quadrant 4  Synchronous Register Clock/Input
I68 pin 68;  " Quadrant 4  Latch Enable/Input
"   **** Quadrant I ****
" I/O LOGIC CELL
" ==============
IO4,IO5,IO6,IO7,IO8,IO9                       pin 4,5,6,7,8,9 istype 'buffer,reg_d';
STF4,STF5,STF6,STF7,STF8,STF9                 node 69,70,71,72,73,74;
IO4Q1,IO5Q1,IO6Q1,IO7Q1,IO8Q1,IO9Q1           node 761,762,763,764,765,766;
IO4Q2,IO5Q2,IO6Q2,IO7Q2,IO8Q2,IO9Q2           node 121,122,123,124,125,126;
IO10,IO11,IO12,IO13,IO14,IO15,IO17            pin 10,11,12,13,14,15,17 istype
                                               'buffer,reg_d'
STF10,STF11,STF12,STF13,STF14,STF15,STF17     node 75,76,77,78,79,80,81;
IO10Q1,IO11Q1,IO12Q1,IO13Q1,IO14Q1,IO15Q1,IO17Q1   node 767,768,769,770,771,772,773;
IO10Q2,IO11Q2,IO12Q2,IO13Q2,IO14Q2,IO15Q2,IO17Q2   node 127,128,129,130,131,132,133;
" INPUT LATCHES
" ===============
IL4,IL5,IL6,IL7,IL8,IL9                       node 813,814,815,816,817,818;
IL10,IL11,IL12,IL13,IL14,IL15,IL17            node 819,820,821,822,823,824,825;
" BURIED LOGIC CELL
" =================
BLC18,BLC19,BLC20,BLC21,BLC22,BLC23           node 178,177,176,175,174,173 ;
declarations "   **** Quadrant II ****
" I/O LOGIC CELL
" ==============
IO18,IO19,IO21,IO22,IO23,IO24                 pin 18,19,21,22,23,24 istype
                                               'buffer,reg_d';
STF18,STF19,STF21,STF22,STF23,STF24           node 82,83,84,85,86,87;
IO18Q1,IO19Q1,IO21Q1,IO22Q1,IO23Q1,IO24Q1     node 774,775,776,777,778,779;
IO18Q2,IO19Q2,IO21Q2,IO22Q2,IO23Q2,IO24Q2     node 134,135,136,137,138,139;
IO25,IO26,IO27,IO28,IO29,IO30,IO31            pin 25,26,27,28,29,30,31 ISTYPE 'BUFFER';
STF25,STF26,STF27,STF28,STF29,STF30,STF31     node 88,89,90,91,92,93,94;
IO25Q1,IO26Q1,IO27Q1,IO28Q1,IO29Q1,IO30Q1,IO31Q1   node 780,781,782,783,784,785,786 ;
IO25Q2,IO26Q2,IO27Q2,IO28Q2,IO29Q2,IO30Q2,IO31Q2   node 140,141,142,143,144,145,146 ;
" INPUT LATCHES
" ===============
IL18,IL19,IL21,IL22,IL23,IL24                 node 826,827,828,829,830,831;
IL25,IL26,IL27,IL28,IL29,IL30,IL31            node 832,833,834,835,836,837,838;
" BURIED LOGIC CELL
" =================
BLC12,BLC13,BLC14,BLC15,BLC16,BLC17           node 184,183,182,181,180,179;
```

5

```
declarations "    **** Quadrant III ****
" I/O LOGIC CELL
" ==============
IO38,IO39,IO40,IO41,IO42,IO43                      pin 38,39,40,41,42,43;
STF38,STF39,STF40,STF41,STF42,STF43                node 95,96,97,98,99,100;
IO38Q1,IO39Q1,IO40Q1,IO41Q1,IO42Q1,IO43Q1          node 787,788,789,790,791,792;
IO38Q2,IO39Q2,IO40Q2,IO41Q2,IO42Q2,IO43Q2          node 147,148,149,150,151,152
                                                    istype'reg_t';
IO44,IO45,IO46,IO47,IO48,IO49,IO51                 pin 44,45,46,47,48,49,51;
STF44,STF45,STF46,STF47,STF48,STF49,STF51          node 101,102,103,104,105,106,107;
IO44Q1,IO45Q1,IO46Q1,IO47Q1,IO48Q1,IO49Q1,IO51Q1   node 793,794,795,796,797,798,799;
IO44Q2,IO45Q2,IO46Q2,IO47Q2,IO48Q2,IO49Q2,IO51Q2   node 153,154,155,156,157,158,159;
" INPUT LATCHES
" ===============
IL38,IL39,IL40,IL41,IL42,IL43                      node 839,840,841,842,843,844;
IL44,IL45,IL46,IL47,IL48,IL49,IL51                 node 845,846,847,848,849,850,851;
" BURIED LOGIC CELL
" =================
BLC6,BLC7,BLC8,BLC9,BLC10,BLC11                    node 190,189,188,187,186,185
                                                    ISTYPE 'REG_D';
declarations "    **** Quadrant IV ****
" I/O LOGIC CELL
" ==============
IO52,IO53,IO55,IO56,IO57,IO58                      pin 52,53,55,56,57,58 istype 'buffer';
STF52,STF53,STF55,STF56,STF57,STF58                node 108,109,110,111,112,113;
IO52Q1,IO53Q1,IO55Q1,IO56Q1,IO57Q1,IO58Q1          node 800,801,802,803,804,805;
IO52Q2,IO53Q2,IO55Q2,IO56Q2,IO57Q2,IO58Q2          node 160,161,162,163,164,165;
IO59,IO60,IO61,IO62,IO63,IO64,IO65                 pin 59,60,61,62,63,64,65 istype 'buffer';
STF59,STF60,STF61,STF62,STF63,STF64,STF65          node 114,115,116,117,118,119,120;
IO59Q1,IO60Q1,IO61Q1,IO62Q1,IO63Q1,IO64Q1,IO65Q1   node 806,807,808,809,810,811,812;
IO59Q2,IO60Q2,IO61Q2,IO62Q2,IO63Q2,IO64Q2,IO65Q2   node 166,167,168,169,170,171,172;
" INPUT LATCHES
" ===============
IL52,IL53,IL55,IL56,IL57,IL58                      node 852,853,854,855,856,857;
IL59,IL60,IL61,IL62,IL63,IL64,IL65                 node 858,859,860,861,862,863,864;
" BURIED LOGIC CELL
" =================
BLC0,BLC1,BLC2,BLC3,BLC4,BLC5                      node 196,195,194,193,192,191;
   H,L,C,D,K,U,X,Z = 1,0,.C.,.D.,.K.,.U.,.X.,.Z.;
"   MACRO (INPUT_LATCH)
INPUT_LATCH MACRO (IL,IO,QUAD_LE)
{?IL.D  = ?IO;
?IL.LE  = ?QUAD_LE;}
equations
"         INPUT LATCH
"  Each of the 52 I/Os has an input latch. When the quadrant latch enable is high,
"  the pin value is latched.
"  When the quadrant latch clock is low, the latch becomes transparent.
IL4.D   = IO4;   "The .D and the .LE (latch enable) extentions are required to describe a
IL4.LE  = I1;    "latch in Abel AHDL. The only allowed input to he latch is the IO pin it
                 "associates with. Theonly allowed .LE input is the quadrant latch enable.
                 "They are Pin 1, 34, 35, and 68.
"  INPUT_LATCH macro is another way to describe the latch.
INPUT_LATCH (IL5,IO5,I1);  "Latch Pin 5
INPUT_LATCH (IL6,IO6,I1);  "Latch Pin 6
IO4.d   = !IO4.fb;  " When .oe is enabled by IO8, IO4 outputs a 1 bit counter.
IO4.oe  = IO8;      " When .oe is disabled by !IO8, IO4 latches a data bit  from the bus.
IO4.ck  = 1;
IO4.ce  = I2;
IO7     = IL4;
test_vectors (   "Test the latches...
[I1,I2,IO8,IO4]-[IO4,IO7])
[ 0, 0, 1,X ]  ->  [  0,   0 ]; "Transparent
```

```
[ 0, C, 1,X  ]  -> [  1,   1 ]; "Transparent
[ U, 0, 0,0  ]  -> [  Z,   0 ]; "Disable .oe and latch 0
[ 1, C, 1,X  ]  -> [  0,   0 ]; "Latched 0
[ 1, C, 1,X  ]  -> [  1,   0 ]; "Latched 0
[ D, C, 1,X  ]  -> [  0,   0 ]; "Transparent
[ 0, C, 1,X  ]  -> [  1,   1 ]; "Transparent
[ 0, C, 1,X  ]  -> [  0,   0 ]; "Transparent
[ U, 0, 0,1  ]  -> [  Z,   1 ]; "Disable .oe and latch 1
[ 1, 0, 1,X  ]  -> [  0,   1 ]; "Latched 1
[ 1, C, 1,X  ]  -> [  1,   1 ]; "Latched 1
[ 1, C, 1,X  ]  -> [  0,   1 ]; "Latched 1
equations
"   CLOCKING OPTIONS
" There are different methods of clocking the registers in the ATV5000. The clock is
" best described as either the AND function of (Quadrant Clock & Clock Product term)
" or the product term by it self. In the following examples the register can be a
"  name from a pin, Q1, Q2, or Buried Logic Cell. The register can be either .t or
"  .d flip-flop.
IO22.d  = !IO22.fb;
IO22.ck = IO18 & !IO19;      "Note there is no .ce equation defined. This is an
IO22.ar = IO21;              "asynchronous clocking method where the quadrant clock
                             "has no effect.
test_vectors (
[IO18,IO19,IO21  ]  -> [ IO22 ])
[  0,   0,1      ]  -> [    0 ];
[  0,   0,0      ]  -> [    0 ];
[  C,   0,0      ]  -> [    1 ];
[  C,   0,0      ]  -> [    0 ];
equations
IO23.d  = !IO23.fb;
IO23.ck = IO18 & !IO19;      "The clock product term (.ck) is used to gate the quadrant
IO23.ce = I32;               "clock pin. Using quadrant clock pin in a synchronous mode
IO23.ar = IO21;              "allows higher clock rate. Pin 32 is the Quadrant 2 clock pin.
test_vectors (
[I32,IO18,IO19,IO21 ]  ->   [ IO23])
[  0,   1,0, 1      ]  ->   [ 0];
[  C,   0,0, 0      ]  ->   [ 0]; "Product term blocks quadrant clock
[  C,   1,0, 0      ]  ->   [ 1]; "Product term enables quadrant clock
[  C,   1,0, 0      ]  ->   [ 0];
equations
IO24.d  = !IO24.fb;
IO24.ck = 1;         "Note that the .ck is defined to 1. The quadrant clock is the only
IO24.ce = I32;       "clocking element. The .ce is Pin 32 because it's the proper quadrant
IO24.ar = IO21;      "clock to use for synchronous operation.
test_vectors (
[ I32, IO21  ]  -> [ IO24  ])
[  0,   1    ]  -> [  0    ];
[  C,   0    ]  -> [  1    ];
[  C,   0    ]  -> [  0    ];
[  C,   0    ]  -> [  1    ];
"  D-TYPE and T-TYPE REGISTERS
"  All the registers in the ATV5000 can be configured as D or T type. The ISTYPE
"  statement after the PIN or NODE definition tells ABEL which type of register is
"  needed.
"  ABC node 122 istype 'reg_t'
"  XYZ pin   5 istype 'reg_d';
"  Use the .t extension for ABC and .d extension for XYZ when you write the equations.
equations
" 3 Bit Synchronous Counter using T flip-flops
IO38Q2.t = 1;                IO38Q2.ck = 1;   IO38Q2.ce = I36;    IO38Q2.ar = IO38;
IO39Q2.t = IO38Q2;           IO39Q2.ck = 1;   IO39Q2.ce = I36;    IO39Q2.ar = IO38;
IO40Q2.t = IO38Q2 & IO39Q2;  IO40Q2.ck = 1;   IO40Q2.ce = I36;    IO40Q2.ar = IO38;
```

```
test_vectors (
[ I36, IO38   ]   -> [ IO40Q2,IO39Q2,IO38Q2])
[   0,   1    ]   -> [ 0, 0, 0 ];
[   C,   0    ]   -> [ 0, 0, 1 ];
[   C,   0    ]   -> [ 0, 1, 0 ];
[   C,   0    ]   -> [ 0, 1, 1 ];
[   C,   0    ]   -> [ 1, 0, 0 ];
[   C,   0    ]   -> [ 1, 0, 1 ];
[   C,   0    ]   -> [ 1, 1, 0 ];
[   C,   0    ]   -> [ 1, 1, 1 ];
[   C,   0    ]   -> [ 0, 0, 0 ];
equations
" 3 Bit Synchronous Counter using D flip-flops
BLC6.d = !BLC6;             BLC6.ck = 1;   BLC6.ce = I36;   BLC6.ar = IO38;
BLC7.d = BLC6 $ BLC7;       BLC7.ck = 1;   BLC7.ce = I36;   BLC7.ar = IO38;
BLC8.d = BLC8 $ (BLC6 & BLC7); BLC8.ck = 1; BLC8.ce = I36;  BLC8.ar = IO38;
test_vectors (
[ I36, IO38   ]   -> [BLC8,BLC7,BLC6])
[   0,   1 ]  -> [  0, 0,  0 ];
[   C,   0 ]  -> [  0, 0,  1 ];
[   C,   0 ]  -> [  0, 1,  0 ];
[   C,   0 ]  -> [  0, 1,  1 ];
[   C,   0 ]  -> [  1, 0,  0 ];
[   C,   0 ]  -> [  1, 0,  1 ];
[   C,   0 ]  -> [  1, 1,  0 ];
[   C,   0 ]  -> [  1, 1,  1 ];
[   C,   0 ]  -> [  0, 0,  0 ];
"       UNIVERSAL AND REGIONAL PRODUCT TERMS
"  A Regional product term has inputs from all the feedbacks of its quadrant Buried
"  Logic Cells, Q1's, Q2's, and the dedicated Inputs. In addition, a Universal
"  product term has the Pin/B Sum Term Feedbacks.
"       OUTPUTS and FEEDBACKS
"  Combinatorial:
"  Use IO pin name to define the equation. The sum terms may combine to allow 5, 9,
"  or 13 product terms depending on the need of the reduced equation. If the reduced
"  equation requires 5 or less product terms, it leaves Q1 and Q2 each with 4 product
"  terms to use.
"  IO52              =  up to 5 PRODUCT TERMS (2 UNIVERSAL)
"  IO52.oe           =  1 PRODUCT TERM (1 UNIVERSAL)
"  IO52Q1.(d or t)   =  up to 4 PRODUCT TERMS (1 UNIVERSAL)
"  IO52Q1.ck         =  1 PRODUCT TERM (1 UNIVERSAL)
"  IO52Q1.ce         =  QUADRANT CLOCK
"  IO52Q1.ar         =  1 PRODUCT TERM (1 UNIVERSAL)
"  IO52Q1.ap         =  1 PRODUCT TERM (1 UNIVERSAL)
"  IO52Q2.(d or t)   =  up to 4 PRODUCT TERMS (1 UNIVERSAL)
"  IO52Q2.ck         =  1 PRODUCT TERM (1 UNIVERSAL)
"  IO52Q2.ce         =  QUADRANT CLOCK
"  IO52Q2.ar         =  1 PRODUCT TERM (1 UNIVERSAL)
"  IO52Q2.ap         =  1 PRODUCT TERM (0 UNIVERSAL)
"  ALLOWED FEEDBACKS:      SOURCE:
"  IO52                --  Pin feedback, after the buffer/inverter.
"  IO52Q1,IO52Q1.FB    --  Q1 register feedback.
"  IO52Q2,IO52Q2.FB    --  Q2 register feedback.
"  Combinatorial:
"  If the reduced equation requires 6 to 9 product terms, it leaves Q2 with 4 product
"  terms to use. Q1 will  feedback the A Sum Term portion of the output logic.
"  IO52              =  up to 9 PRODUCT TERMS (3 UNIVERSAL)
"  IO52.oe           =  1 PRODUCT TERM (1 UNIVERSAL)
"  IO52Q2.(d or t)   =  up to 4 PRODUCT TERMS (1 UNIVERSAL)
"  IO52Q2.ck         =  1 PRODUCT TERM (1 UNIVERSAL)
"  IO52Q2.ce         =  QUADRANT CLOCK
"  IO52Q2.ar         =  1 PRODUCT TERM (1 UNIVERSAL)
"  IO52Q2.ap         =  1 PRODUCT TERM (0 UNIVERSAL)
```

```
"  ALLOWED FEEDBACKS:                         SOURCE:
"  IO52,IO52.PIN         --                   Pin feedback, after the buffer/inverter.
"  IO52Q2,IO52Q2.FB,IO52Q2.Q --               Q2 register feedback.
"  Combinatorial:
"  If the reduced equation requires more than 9 product terms, it leaves no product
"  terms for Q1 and Q2. Q1 feeds back A Sum Term and Q2 feeds back C Sum Term
"  IO52            =   up to 13 PRODUCT TERMS (4 UNIVERSAL)
"  IO52.oe         =   1 PRODUCT TERM (1 UNIVERSAL)
"  ALLOWED FEEDBACKS:      SOURCE:
"  IO52            --   Pin feedback, after the buffer/inverter.
"  Registered:
"  Use IO pin name to define the equation. The sum terms may combine to allow 4, 9,
"  or 13 product terms depending on the need of the reduced equation. If the reduced
"  equation requires 4 or less product terms, you may define a 5 product term
"  equation for the STF (Sum Term Feedback) and define a 4 product term equation for
"  Q2.
"  IO52.(d or t)   =   up to 4 PRODUCT TERMS (1 UNIVERSAL)
"  IO52.oe         =   1 PRODUCT TERM (1 UNIVERSAL)
"  IO52.ck         =   1 PRODUCT TERM (1 UNIVERSAL)
"  IO52.ce         =   QUADRANT CLOCK
"  IO52.ar         =   1 PRODUCT TERM (1 UNIVERSAL)
"  IO52.ap         =   1 PRODUCT TERM (1 UNIVERSAL)
"  STF52           =   up to 5 PRODUCT TERMS (2 UNIVERSAL)
"  IO52Q2.(d or t) =   up to 4 PRODUCT TERMS (1 UNIVERSAL)
"  IO52Q2.ck       =   1 PRODUCT TERM (1 UNIVERSAL)
"  IO52Q2.ce       =   QUADRANT CLOCK
"  IO52Q2.ar       =   1 PRODUCT TERM (1 UNIVERSAL)
"  IO52Q2.ap       =   1 PRODUCT TERM (0 UNIVERSAL)

"  ALLOWED FEEDBACKS:                SOURCE:
"  IO52.FB                   --      Register feedback prior to buffer/inverter.
"  IO52Q2,IO52Q2.FB          --      Q2 register feedback.
"  STF52                     --      Sum Term Feedback for logic expansion.
"  Registered:
"  If the reduced equation requires 9 to 5 product terms, you may write a 4 product
"  term equation for Q2.
"  IO52.(d or t)   =   up to 9 PRODUCT TERMS (3 UNIVERSAL)
"  IO52.oe         =   1 PRODUCT TERM (1 UNIVERSAL)
"  IO52.ck         =   1 PRODUCT TERM (1 UNIVERSAL)
"  IO52.ce         =   QUADRANT CLOCK
"  IO52.ar         =   1 PRODUCT TERM (1 UNIVERSAL)
"  IO52.ap         =   1 PRODUCT TERM (1 UNIVERSAL)
"  IO52Q2.(d or t) =   up to 4 PRODUCT TERMS (1 UNIVERSAL)
"  IO52Q2.ck       =   1 PRODUCT TERM (1 UNIVERSAL)
"  IO52Q2.ce       =   QUADRANT CLOCK
"  IO52Q2.ar       =   1 PRODUCT TERM (1 UNIVERSAL)
"  IO52Q2.ap       =   1 PRODUCT TERM (0 UNIVERSAL)
"  ALLOWED FEEDBACKS:              SOURCE:
"  IO52                    --      Pin feedback, after the buffer/inverter.
"  IO52.FB                 --      Register feedback prior to buffer/inverter.
"  IO52Q2,IO52Q2.FB,       --      Q2 register feedback.
"  Registered:
"  If the reduced equation requires more than 9 product terms, it leaves zero
"  product terms for Q1 and Q2.
"  IO52.(d or t)   =   up to 13 PRODUCT TERMS (4 UNIVERSAL)
"  IO52.oe         =   1 PRODUCT TERM (1 UNIVERSAL)
"  IO52.ck         =   1 PRODUCT TERM (1 UNIVERSAL)
"  IO52.ce         =   QUADRANT CLOCK
"  IO52.ar         =   1 PRODUCT TERM (1 UNIVERSAL)
"  IO52.ap         =   1 PRODUCT TERM (1 UNIVERSAL)
"  ALLOWED FEEDBACKS:              SOURCE:
"  IO52,                   --      Pin feedback, after the buffer/inverter.
"  IO52.FB                 --      Register feedback prior to buffer/inverter.
```

5

```
"      AP, AR, and CK
"  Each of the 128 registers has its own .AP (Asynchronous Preset), .AR (Asynchronous
"  Reset) and .CK. All ARs and CKs are Universal product terms. The APs are Universal
"  product terms except the APs for Q2.
"  NOTE: AP and AR should never be active at the same time.
equations
IO10.d     =   !IO10.fb;
IO10.ck    =   IO8;
IO10.ar    =   IO11;
IO10.ap    =   IO12;
test_vectors (
[ IO8, IO11, IO12     ]  ->   [ IO10])
[      0, 1, 0        ]  ->   [  0  ]; "AR
[      0, 0, 0        ]  ->   [  0  ];
[      C, 0, 0        ]  ->   [  1  ];
[      C, 0, 0        ]  ->   [  0  ];
[      0, 0, 1        ]  ->   [  1  ]; "AP
[      0, 1, 0        ]  ->   [  0  ]; "AR
"       BURIED LOGIC CELLS
"  Buried Logic Cell can be configured as a register feedback like Q2. It can also be
"  configured as combinatorial feedback to accommodate for logic expansion.
equations
BLC17.t    =   1; "Registered
BLC17.ck   =   IO9;
BLC17.ar   =   IO21;
BLC14      =   I1; "Combinatiorial
IO25       =   BLC17;
IO26       =   BLC14;
test_vectors (
[IO9, IO21, I1   ]   -> [ BLC17,BLC14,IO25,IO26])
[  0, 1, 0       ]   -> [ 0, 0, 0, 0 ];
[  C, 0, 0       ]   -> [ 1, 0, 1, 0 ];
[  0, 0, 1       ]   -> [ 1, 1, 1, 1 ];
[  C, 0, 1       ]   -> [ 0, 1, 0, 1 ];
[  C, 0, 0       ]   -> [ 1, 0, 1, 0 ];
END;
```

Example Cupl™ Description File

```
Name      V5K;
Partno    NA;
Date      7/22/91;
Revision  01;
Designer  Joe Yu;
Company Atmel Corporation;
Assembly  NA;
Location  U1;
Device    V5000;
Format    j;
/* Compiled with Cupl 386 version */
/* The IOs, registers, inputs, and combinatorial sum terms feedbacks are named with
 * the following prefixes for clarity:
 * Prefix
 * I       -  Quadrant Clocks, Latch Enables.
 * IO      -  IO pins
 * BLC     -  Buried Logic Cells
 * STF     -  Sum Term Feedbacks
 * Valid CUPL HDL identifiers can be used in place of them. */
pin 1   =    I1;    /* Quadrant 1           Latch Enable/Input                   */
pin 2   =    I2;    /* Quadrant 1           Synchronous Register Clock/Input */
pin 32  =    I32;   /* Quadrant 2           Synchronous Register Clock/Input */
pin 34  =    I34;   /* Quadrant 2           Latch Enable/Input                   */
pin 35  =    I35;   /* Quadrant 3           Latch Enable/Input                   */
pin 36  =    I36;   /* Quadrant 3           Synchronous Register Clock/Input */
pin 66  =    I66;   /* Quadrant 4           Synchronous Register Clock/Input */
pin 68  =    I68;   /* Quadrant 4           Latch Enable/Input                   */
          /**** Quadrant I ****/
 /*  I/O LOGIC CELL    */
 /*  ==============    */
pin     [4..9]        =   [IO4..9];
pinnode [209..204]    =   [STF4..9];
pinnode [157..152]    =   [IO4Q1,IO5Q1,IO6Q1,IO7Q1,IO8Q1,IO9Q1];
pinnode [105..100]    =   [IO4Q2,IO5Q2,IO6Q2,IO7Q2,IO8Q2,IO9Q2];
pin     [10..15,17]   =   [IO10..15,IO17];
pinnode [203..197]    =   [STF10..15,STF17];
pinnode [151..145]    =   [IO10Q1,IO11Q1,IO12Q1,IO13Q1,IO14Q1,IO15Q1,IO17Q1];
pinnode [99..93]      =   [IO10Q2,IO11Q2,IO12Q2,IO13Q2,IO14Q2,IO15Q2,IO17Q2];
/* BURIED LOGIC CELL     */
/* =================     */
pinnode     [69..74]  =   [BLC18..23];
          /**** Quadrant II ****/
/* I/O LOGIC CELL */
/* ==============    */
pin     [18,19,21..24]  =   [IO18,IO19,IO21..24];
pinnode [210..215]    =   [STF18,STF19,STF21..24];
pinnode [158..163]    =   [IO18Q1,IO19Q1,IO21Q1,IO22Q1,IO23Q1,IO24Q1];
pinnode [106..111]    =   [IO18Q2,IO19Q2,IO21Q2,IO22Q2,IO23Q2,IO24Q2];
pin     [25..31]      =   [IO25..31];
pinnode [216..222]    =   [STF25..31];
pinnode [164..170]    =   [IO25Q1,IO26Q1,IO27Q1,IO28Q1,IO29Q1,IO30Q1,IO31Q1];
pinnode [112..118]    =   [IO25Q2,IO26Q2,IO27Q2,IO28Q2,IO29Q2,IO30Q2,IO31Q2];
/* BURIED LOGIC CELL     */
/* =================     */
pinnode [80..75]      =   [BLC12..17];
          /**** Quadrant III ****/
/* I/O LOGIC CELL    */
/* ==============*/
pin     [38..43]      =   [IO38..43];
pinnode [235..230]    =   [STF38..43];
```

5

```
pinnode [183..178]      =     [IO38Q1,IO39Q1,IO40Q1,IO41Q1,IO42Q1,IO43Q1];
pinnode [131..126]      =     [IO38Q2,IO39Q2,IO40Q2,IO41Q2,IO42Q2,IO43Q2];
pin     [44..49,51]     =     [IO44..49,IO51];
pinnode [229..223]      =     [STF44..49,STF51];
pinnode [177..171]      =     [IO44Q1,IO45Q1,IO46Q1,IO47Q1,IO48Q1,IO49Q1,IO51Q1];
pinnode [125..119]      =     [IO44Q2,IO45Q2,IO46Q2,IO47Q2,IO48Q2,IO49Q2,IO51Q2];
/* BURIED LOGIC CELL    */
/* =================    */
pinnode [81..86]        =     [BLC6..11];
            /**** Quadrant IV ****/
/* I/O LOGIC CELL    */
/* ==============    */
pin     [52,53,55..58]  =     [IO52,IO53,IO55..58];
pinnode [236..241]      =     [STF52,STF53,STF55..58];
pinnode [184..189]      =     [IO52Q1,IO53Q1,IO55Q1,IO56Q1,IO57Q1,IO58Q1];
pinnode [132..137]      =     [IO52Q2,IO53Q2,IO55Q2,IO56Q2,IO57Q2,IO58Q2];
pin     [59..65]        =     [IO59..65];
pinnode [242..248]      =     [STF59..65];
pinnode [190..196]      =     [IO59Q1,IO60Q1,IO61Q1,IO62Q1,IO63Q1,IO64Q1,IO65Q1];
pinnode [138..144]      =     [IO59Q2,IO60Q2,IO61Q2,IO62Q2,IO63Q2,IO64Q2,IO65Q2];
/* BURIED LOGIC CELL    */
/* =================    */
pinnode [92..87]        =     [BLC0..5];
/*          INPUT LATCH
   Each of the 52 I/Os has an input latch. When the quadrant latch enable is high,
   the pin value is latched. When the quadrant latch clock is low, the latch becomes
   transparent.  */
IO4.d   = !IO4;
IO4.oe  = IO8.io;
IO4.ce  = 'B'1;
IO7     = IO4.IOL;  /* When the latched input is needed in the design, i.e.. IO4.IOL, IO4
                       is configured as having a latched input.        */
/*          CLOCKING OPTIONS
   There are different methods of clocking the registers in the ATV5000. The clock is
   best described as either the AND function of (Quadrant Clock & Clock Product term)
   or the product term by it self. In the following examples the register can be a
   name from a pin, Q1, Q2, or Buried Logic Cell. The register can be either  .t or .d
   flip-flop.   */
IO22.d  = !IO22;
IO22.ck = IO18 & !IO19;/* Note there is no .ce equation defined. This is an     */
IO22.ar = IO21;        /* asynchronous clocking method where the quadrant clock */
                       /* has no effect.                                        */
IO23.d  = !IO23;
IO23.ce = IO18 & !IO19; /* The clock product term is used to gate the quadrant  */
IO23.ar = IO21;        /* clock pin.Using quadrant clock pin in a synchronous   */
IO24.d  = !IO24;       /* mode allows higher clock rate. Pin 32 is the Quadrant */
                       /* 2 clock pin.                                          */
IO24.d  = !IO24;
IO24.ce = 'B'1;        /* Note that the .ce is defined to 1. The quadrant clock */
IO24.ar = IO21;        /* is the only clocking element. The clock is Pin 32     */
                       /* because it's the proper quadrant clock to use for     */
                       /* synchronous operation.                                */
/*          D-TYPE and T-TYPE REGISTERS
   All the registers in the ATV5000 can be configured as D or T type. Use the .d
   extension for D type register and .t extension for T type register.  */
/* 3 Bit Synchronous Counter using T flip-flops */
IO38Q2.t = 'B'1;               IO38Q2.ce = 'B'1;    IO38Q2.ar = IO38;
IO39Q2.t = IO38Q2;             IO39Q2.ce = 'B'1;    IO39Q2.ar = IO38;
IO40Q2.t = IO38Q2 & IO39Q2;    IO40Q2.ce = 'B'1;    IO40Q2.ar = IO38;
/* 3 Bit Synchronous Counter using D flip-flops */
BLC6.d = !BLC6;                BLC6.ce = 'B'1;      BLC6.ar = IO38;
BLC7.d = BLC6 $ BLC7;          BLC7.ce = 'B'1;      BLC7.ar = IO38;
BLC8.d = BLC8 $ (BLC6 & BLC7); BLC8.ce = 'B'1;      BLC8.ar = IO38;
```

```
/*      UNIVERSAL AND REGIONAL PRODUCT TERMS
   A Regional product term has inputs from all the feedbacks of its quadrant Buried
   Logic Cells, Q1's, Q2's, and the dedicated Inputs. In addition, a Universal product
   term has the Pin/B Sum Term Feedbacks.                  */
/*       OUTPUTS and FEEDBACKS
   Combinatorial:
   Use IO pin name to define the equation. The sum terms may combine to allow 5, 9, or
   13 product terms depending on the need of the reduced equation. If the reduced
   equation requires 5 or less product terms, it leaves Q1 and Q2 each with 4 product
   terms to use.
        IO52                 = up to 5 PRODUCT TERMS (2 UNIVERSAL)
        IO52.oe              = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52Q1.(d or t)      = up to 4 PRODUCT TERMS (1 UNIVERSAL)
        IO52Q1.(ck or ce)    = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52Q1.ar            = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52Q1.ap            = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52Q2.(d or t)      = up to 4 PRODUCT TERMS (1 UNIVERSAL)
        IO52Q2.(ck or ce)    = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52Q2.ar            = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52Q2.ap            = 1 PRODUCT TERM (0 UNIVERSAL)
   ALLOWED FEEDBACKS:          SOURCE:
   IO52,IO52.IO         --     Pin feedback, after the buffer/inverter.
   IO52Q1               --     Q1 register feedback.
   IO52Q2               --     Q2 register feedback.           */
/* Combinatorial:
   If the reduced equation requires 6 to 9 product terms, it leaves Q2 with 4 product
   terms to use. Q1 will feedback the A Sum Term portion of the output logic.
        IO52                 = up to 9 PRODUCT TERMS (3 UNIVERSAL)
        IO52.oe              = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52Q2.(d or t)      = up to 4 PRODUCT TERMS (1 UNIVERSAL)
        IO52Q2.(ck or ce)    = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52Q2.ar            = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52Q2.ap            = 1 PRODUCT TERM (0 UNIVERSAL)
   ALLOWED FEEDBACKS:          SOURCE:
   IO52,IO52.IO         --     Pin feedback, after the buffer/inverter.
   IO52Q2               --     Q2 register feedback.           */
/* Combinatorial:
   If the reduced equation requires more than 9 product terms, it leaves no product
   terms for Q1 and Q2. Q1 feeds back A. Sum Term and Q2 feeds back C Sum Term.
        IO52                 = up to 13 PRODUCT TERMS (4 UNIVERSAL)
        IO52.oe              = 1 PRODUCT TERM (1 UNIVERSAL)
   ALLOWED FEEDBACKS:          SOURCE:
   IO52,IO52.IO         --     Pin feedback, after the buffer/inverter.      */
/* Registered:
   Use IO pin name to define the equation. The sum terms may combine to allow 4, 9, or
   13 product terms depending on the  need of the reduced equation. If the reduced
   equation requires 4 or less product terms, you may define a 5 product term equation
   for the STF (Sum Term Feedback) and define a 4 product
   term equation for Q2.
        IO52.(d or t)        = up to 4 PRODUCT TERMS (1 UNIVERSAL)
        IO52.oe              = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52.(ck or ce)      = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52.ar              = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52.ap              = 1 PRODUCT TERM (1 UNIVERSAL)
        STF52 = up to 5 PRODUCT TERMS (2 UNIVERSAL)
        IO52Q2.(d or t)      = up to 4 PRODUCT TERMS (1 UNIVERSAL)
        IO52Q2.(ck or ce)    = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52Q2.ar            = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52Q2.ap            = 1 PRODUCT TERM (0 UNIVERSAL)
   ALLOWED FEEDBACKS:          SOURCE:
   IO52                 --     Register feedback prior to buffer/inverter.
   IO52Q2               --     Q2 register feedback.
   STF52                --     Sum Term Feedback for logic expansion.      */
```

5

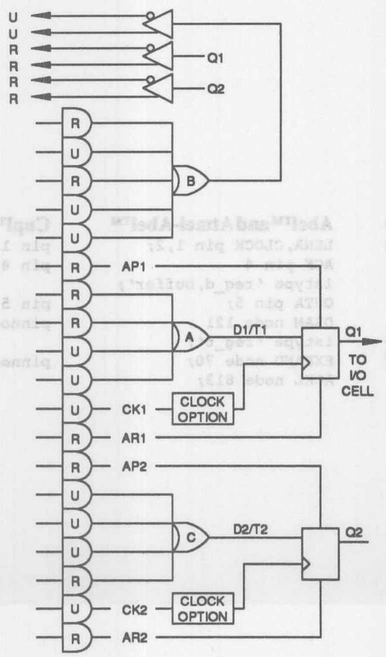
```
/* Registered:
   If the reduced equation requires 9 to 5 product terms, you may write a 4 product
   term equation for Q2.
        IO52.(d or t)       = up to 9 PRODUCT TERMS (3 UNIVERSAL)
        IO52.oe             = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52.(ck or ce)     = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52.ar             = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52.ap             = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52Q2.(d or t)     = up to 4 PRODUCT TERMS (1 UNIVERSAL)
        IO52Q2.(ck or ce)   = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52Q2.ar           = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52Q2.ap           = 1 PRODUCT TERM (0 UNIVERSAL)
   ALLOWED FEEDBACKS:          SOURCE:
   IO52.IO             --      Pin feedback, after the buffer/inverter.
   IO52                --      Register feedback prior to buffer/inverter.
   IO52Q2              --      Q2 register feedback. */
/* Registered:
   If the reduced equation requires more than 9 product terms, it leaves zero product
   terms for Q1 and Q2.
        IO52.(d or t)       = up to 13 PRODUCT TERMS (4 UNIVERSAL)
        IO52.oe             = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52.(ck or ce)     = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52.ar             = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52.ap             = 1 PRODUCT TERM (1 UNIVERSAL)
   ALLOWED FEEDBACKS:          SOURCE:
   IO52.IO             --      Pin feedback, after the buffer/inverter.
   IO52                --      Register feedback prior to buffer/inverter. */
/*         AP, AR, and CK
   Each of the 128 registers has its own .AP (Asynchronous Preset), .AR (Asynchronous
   Reset) and .CK or .CE. All ARs and CKs are Universal product terms. The APs are
   Universal product terms except the APs for Q2.
   NOTE:  AP and AR should never be active at the same time.          */
IO10.d     =  !IO10;
IO10.ck    =  IO8;
IO10.ar    =  IO11;
IO10.ap    =  IO12;
/*         BURIED LOGIC CELLS
   Buried Logic Cell can be configured as a register feedback like Q2. It can also be
   configured as combinatorial feedback to accommodate for logic expansion. */
BLC17.t    =  'B'1;  /*Registered */
BLC17.ck   =  IO9;
BLC17.ar   =  IO21;
BLC14      =  I1; /*Combinatiorial */
IO25       =  BLC17;
IO26       =  BLC14;
END;
```

Using the ATV5100

The ATV5100 logic cells retain the ATV2500's basic features with two registers, three sum terms and Q1, Q2, and pin feedback options. In addition, these following features make the ATV5100 even more versatile:

1. Input latches are added to every I/O pin. Each latch can be enabled or disabled independently.
2. The ATV5100 can be configured to feedback a combinatorial sum term and send Q1 to the output. Feeding the combinatorial sum term internally makes implementing logic equations with large number of product terms possible without sacrificing an I/O pin.
3. Dedicated clock pins and individual clock product terms create multiple clock options for each flip-flop.
4. The addition of the buried logic cells handles more logic than ever.
5. Each flip-flop can be configured as D- or T-type flip-flop.
6. Asynchronous preset, asynchronous reset and programmable output polarity allow registered outputs using fewer product terms.
7. Same architecture as the ATV5000 with more emphasis on universal routing.

Logic Cell with Buried Sum Term and Register to I/O Cell

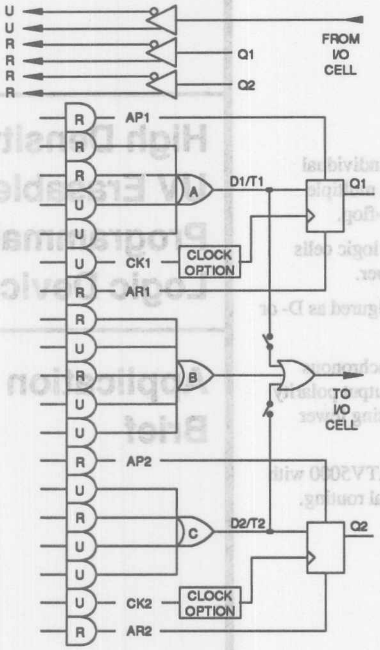


High Density UV Erasable Programmable Logic Device

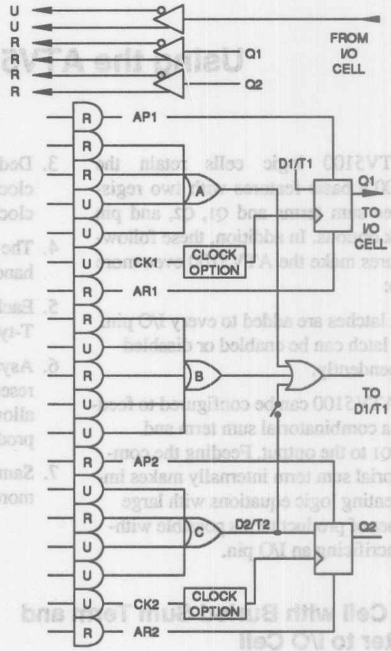
Application Brief

5

Logic Cell, Two Buried Registers, Combinatorial to I/O Cell



Logic Cell with Combinable Sum Terms, Register to I/O Cell



The following are sections of Abel™ and Cupl™ source files to illustrate how each of these features is described in the Abel™ and Cupl™ high level description languages.

Pin and Node Assignments

All the buried registers used in the design need to be assigned node numbers. The following tables show the complete set of node numbers by quadrant.

Abel™ and Atmel-Abel™

```
LENA,CLOCK pin 1,2;
ACK pin 4
istype 'reg_d,buffer';
OUTA pin 5;
DRAM node 121
istype 'reg_t';
EXPAND node 70;
ACKL node 813;
```

Cupl™

```
pin 1,2 = LENA,CLOCK;
pin 4 = ACK;

pin 5 = OUTA;
pinnode 105 = DRAM;

pinnode 208 = EXPAND;
```

ATV5100 Abel™ and Atmel-Abel™ Node Numbers

Quadrant I

Pin	Input Latch	Sum Term A	Sum Term B	Sum Term C
4	813	761	69	121
5	814	762	70	122
6	815	763	71	123
7	816	764	72	124
8	817	765	73	125
9	818	766	74	126
10	819	767	75	127
11	820	768	76	128
12	821	769	77	129
13	822	770	78	130
14	823	771	79	131
15	824	772	80	132
17	825	773	81	133

6 Buried Logic Cells
B23 - B18 node 173 - 178
LCK1 pin 1; Quadrant Latch clock
RCK1 pin 2; Quadrant Synchronous Register Clock

Quadrant II

Pin	Input Latch	Sum Term A	Sum Term B	Sum Term C
18	826	774	82	134
19	827	775	83	135
21	828	776	84	136
22	829	777	85	137
23	830	778	86	138
24	831	779	87	139
25	832	780	88	140
26	833	781	89	141
27	834	782	90	142
28	835	783	91	143
29	836	784	92	144
30	837	785	93	145
31	838	786	94	146

6 Buried Logic Cells
B17 - B12 node 179 - 184
LCK2 pin 34; Quadrant Latch Clock
RCK2 pin 32; Quadrant Synchronous Register Clock

Quadrant III

Pin	Input Latch	Sum Term A	Sum Term B	Sum Term C
38	839	787	95	147
39	840	788	96	148
40	841	789	97	149
41	842	790	98	150
42	843	791	99	151
43	844	792	100	152
44	845	793	101	153
45	846	794	102	154
46	847	795	103	155
47	848	796	104	156
48	849	797	105	157
49	850	798	106	158
51	851	799	107	159

6 Buried Logic Cells
B11 - B6 node 185 - 190
LCK3 pin 35; Quadrant Latch Clock
RCK3 pin 36; Quadrant Synchronous Register Clock

Quadrant IV

Pin	Input Latch	Sum Term A	Sum Term B	Sum Term C
52	852	800	108	160
53	853	801	109	161
55	854	802	110	162
56	855	803	111	163
57	856	804	112	164
58	857	805	113	165
59	858	806	114	166
60	859	807	115	167
61	860	808	116	168
62	861	809	117	169
63	862	810	118	170
64	863	811	119	171
65	864	812	120	172

6 Buried Logic Cells
B5 - B0 node 191 - 196
LCK4 pin 68; Quadrant Latch Clock
RCK4 pin 66; Quadrant Synchronous Register Clock

ATV5100 CupI™ Node Numbers

Quadrant I

Pin	Sum Term A	Sum Term B	Sum Term C
4	157	209	105
5	156	208	104
6	155	207	103
7	154	206	102
8	153	205	101
9	152	204	100
10	151	203	99
11	150	202	98
12	149	201	97
13	148	200	96
14	147	199	95
15	146	198	94
17	145	197	93

6 Buried Logic Cells
B23 - B18 node 74 - 69
LCK1 pin 1; Quadrant Latch Clock
RCK1 pin 2; Quadrant Synchronous Register Clock

Quadrant II

Pin	Sum Term A	Sum Term B	Sum Term C
18	158	210	106
19	159	211	107
21	160	212	108
22	161	213	109
23	162	214	110
24	163	215	111
25	164	216	112
26	165	217	113
27	166	218	114
28	167	219	115
29	168	220	116
30	169	221	117
31	170	222	118

6 Buried Logic Cells
B17 - B12 node 75 - 80
LCK2 pin 34; Quadrant Latch Clock
RCK2 pin 32; Quadrant Synchronous Register Clock

Quadrant III

Pin	Sum Term A	Sum Term B	Sum Term C
38	183	235	131
39	182	234	130
40	181	233	129
41	180	232	128
42	179	231	127
43	178	230	126
44	177	229	125
45	176	228	124
46	175	227	123
47	174	226	122
48	173	225	121
49	172	224	120
51	171	223	119

6 Buried Logic Cells
B11 - B6 node 86 - 81
LCK3 pin 35; Quadrant Latch Clock
RCK3 pin 36; Quadrant Synchronous Register Clock

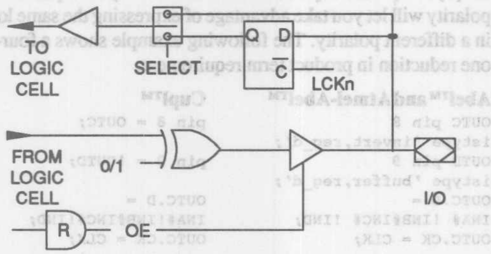
Quadrant IV

Pin	Sum Term A	Sum Term B	Sum Term C
52	184	236	132
53	185	237	133
55	186	238	134
56	187	239	135
57	188	240	136
58	189	241	137
59	190	242	138
60	191	243	139
61	192	244	140
62	193	245	141
63	194	246	142
64	195	247	143
65	196	248	144

6 Buried Logic Cells
B5 - B0 node 87 - 92
LCK4 pin 68; Quadrant Latch Clock
RCK4 pin 66; Quadrant Synchronous Register Clock

Input Latch

Each of the 52 I/Os has an input latch that can be enabled and disabled individually. When the latch is enabled and latch clock is high, the pin value is latched. When the quadrant latch clock is low, the latch becomes transparent. When the latch is disabled, an ATV5100 I/O acts like those of the ATV750 and ATV2500 I/Os. The pin input is fed directly to the array (except when sum term B is being used as a buried feedback).



The .D and the .LE (latch enable) equations are required to enable the input latch in Abel™ AHDL. The only allowed input to the latch is the IO pin with which it is associated. The only allowed .LE input is the quadrant latch clock (pin 1,34, 35, or 68). Notice that Cupl™ does not have node numbers for the input latches. Any pin name used in a feedback with the dot extension IOL tells Cupl™ that particular pin should be a latched pin.

Abel™ and Atmel-Abel™	Cupl™
`ACKL.D = ACK;`	
`ACKL.LE = LENA;`	
`OUTA := ACKL;`	`OUTA.D = ACK.IOL;`

Internal Combinatorial Feedback for an I/O Cell

To implement the combinatorial feedback of the B sum term, first define the node name with the corresponding node number. This node will take a five product term equation. Regular syntax describing a combinatorial equation will describe the B sum term.

Note: This B sum term node number is defined only when this feature is needed. When this feature is used, the output is from Q1 through an inverter/buffer. *The I/O pin becomes a output-only pin. It cannot be used as an input or as an input/output.*

Abel™ and Atmel-Abel™	Cupl™
`OUTA pin 5;`	`pin 5 = OUTA;`
`EXPAND node 70;`	`pinnode 208 = EXPAND;`
`OUTA.D =`	`OUTA.D =`
`INA&INB# !INC&!IND;`	`INA&INB#!INC&!IND;`
`EXPAND = INA & !INC # IND;`	`EXPAND = INA & !INC # IND;`

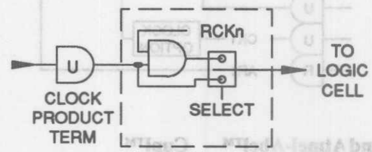
(OUTA and EXPAND are in the same I/O logic cell. OUTA is the Q1 output to pin after the inverter/buffer and EXPAND is the combinatorial sum term feedback. The feedbacks from this logic cell are Q1 before the inverter/buffer, Q2, and the B sum term)

Clocking Options

There are different methods of clocking the registers in the ATV5100. The clock options can best described as either the AND function of (quadrant clock & clock product term) or purely the function of the clock product term (like ATV750 and ATV2500).

Synchronous Operation

The quadrant clock is the only clocking element in this mode of operation. The clock product term must be defined to be equal to 1. In Abel AHDL, .CK defines the clock product term and .CE



is the corresponding quadrant clock.

In Cupl™, .CE has a different implication. The keyword .CE means the user wants the AND function (quadrant clock & clock product term) for the clock.

Abel™ and Atmel-Abel™	Cupl™
`OUTA.CK = 1;`	
`OUTA.CE = CLOCK; "CLOCK is pin 2`	`OUTA.CE = 'B'1;`

Gated Synchronous Operation

This clock option still uses the fast quadrant register clock pin, but now it has a gating element. The clock product term enables or disables the clock going to the register.

Abel™ and Atmel-Abel™	Cupl™
`OUTA.CK = INX & INY & !INZ;`	
`OUTA.CE = CLOCK; "CLOCK is pin 2`	`OUTA.CE = INX&INY&!INZ;`

Asynchronous Operation

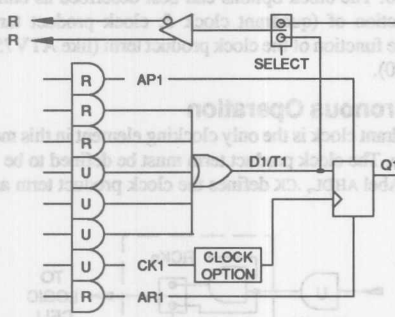
The quadrant clock has no effect on the register in this mode of clock option. The register is clocked by the clock product term like the ATV750 and ATV2500 registers.

Abel™ and Atmel-Abel™	Cupl™
`OUTA.CK =`	`OUTA.CK =`
`INA & !INB & INC;`	`INA & !INB & INC;`

5

Buried Logic Cells

There are six buried logic cells in each quadrant. Every buried logic cell can be configured as a buried register or as a combinatorial feedback to the quadrant array. In the buried register mode, it's used the same way as Q1 and Q2. In the combinatorial feedback mode, it's used the same way as the B sum term feedback.



Abel™ and Atmel-Abel™	Cupl™
`ADDR,XDATA node 173,174;`	`pinnode 74,73 = ADDR,XDATA;`
`ADDR = A16&A15&!A14#RST;`	`ADDR = A16&A15&!A14#RST;`
`XDATA.T = !A16 & RST;`	`XDATA.T = !A16 & RST;`
`XDATA.CK = INA & INC;`	`XDATA.CK = INA & INC;`
`XDATA.AR = INB;`	`XDATA.AR = INB;`

D-Type and T-Type Registers

All the registers in the ATV5100 can be configured as D- or T-type. The ISTYPE statement after the PIN or NODE definition in Abel™ AHDL tells Abel™ which type of register is needed.

Abel™ and Atmel-Abel™	Cupl™
`ABC node 123 istype 'reg_t';`	`pinnode 103 = ABC;`
`XYZ pin 7 istype 'reg_d';`	`pin 7 = XYZ;`
`ABC.T = INA & INB # INC;`	`ABC.T = INA & INB # INC;`
`XYZ.D = !INA # INC;`	`XYZ.D = !INA # INC;`

In both languages, the .T extension is reserved for T flip-flop and .D extension for D flip-flop.

Asynchronous Preset and Reset

One of the advantanges of having a programmable output polarity I/O is that the same combinatorial output often takes fewer product terms to implement using negative logic rather than positive logic or vice versa.

The same can be done for a registered output provided the power-up state is not crucial (all flip-flops are reset upon power up). With an asynchronous preset and an asynchronous reset, simply swap the two product terms and invert your equation and polarity will let you take advantage of expressing the same logic in a different polarity. The following example shows a four-to-one reduction in product term requirement.

Abel™ and Atmel-Abel™	Cupl™
`OUTC pin 8`	`pin 8 = OUTC;`
`istype 'invert,reg_d';`	
`OUTD pin 9`	`pin 9 = !OUTD;`
`istype 'buffer,reg_d';`	
`OUTC.D =`	`OUTC.D =`
`INA# !INB#INC# !IND;`	`INA#!INB#INC#!IND;`
`OUTC.CK = CLK;`	`OUTC.CK = CLK;`
`OUTC.AR = !INA & RST;`	`OUTC.AR = !INA & RST;`
`OUTC.AP = INB & PRESET;`	`OUTC.AP = INB & PRESET;`
`OUTD.D =`	`OUTD.D =`
`!INA&INB&!INC&IND;`	`!INA&INB&!INC&IND;`
`OUTD.CK = CLK;`	`OUTD.CK = CLK;`
`OUTD.AR = INB & PRESET;`	`OUTD.AR = INB & PRESET;`
`OUTD.AP = !INA & RST;`	`OUTD.AP = !INA & RST;`

In the ATV5100, all asynchronous reset and preset terms are regional. Hence, their inputs must be from the eight dedicated input pins or from the other regional bus signals (logic cell Q1 and Q2 feedback and buried cell feedback).

Abel™ and Cupl™ may be trademarks of others.

Example Abel™ Description File

```
module  V5100;
title   'Demo ATV5100 features with ATmel-abel (IBM386 or compatible)
         Atmel Corporation PLD      Joe Yu          May 20, 1992'
V5K device 'P5100';
"  The IOs, registers, latches, inputs, and combinatorial sum terms feedbacks are named "with the
following prefixes for clarity:
"   Prefix
"   I       -   Quadrant Clocks, Latch Enables.
"   IO      -   IO pins
"   IL      -   Input Latches
"   BLC     -   Buried Logic Cells
"   STF     -   Sum Term Feedbacks
"   Valid Abel AHDL identifiers can be used in place of them.
declarations
I1 pin 1;       " Quadrant 1   Latch Enable/Input
I2 pin 2;       " Quadrant 1   Synchronous Register Clock/Input
I32 pin 32;     " Quadrant 2   Synchronous Register Clock/Input
I34 pin 34;     " Quadrant 2   Latch Enable/Input
I35 pin 35;     "Quadrant 3    Latch Enable/Input
I36 pin 36;     "Quadrant 3    Synchronous Register Clock/Input
I66 pin 66;     "Quadrant 4    Synchronous Register Clock/Input
I68 pin 68;     "Quadrant 4    Latch Enable/Input
"   **** Quadrant I ****
" I/O LOGIC CELL
" ===============
IO4,IO5,IO6,IO7,IO8,IO9                           pin 4,5,6,7,8,9 istype 'buffer,reg_d';
STF4,STF5,STF6,STF7,STF8,STF9                     node 69,70,71,72,73,74;
IO4Q1,IO5Q1,IO6Q1,IO7Q1,IO8Q1,IO9Q1               node 761,762,763,764,765,766 ;
IO4Q2,IO5Q2,IO6Q2,IO7Q2,IO8Q2,IO9Q2               node 121,122,123,124,125,126 ;
IO10,IO11,IO12,IO13,IO14,IO15,IO17                pin 10,11,12,13,14,15,17 istype 'buffer,reg_d';
STF10,STF11,STF12,STF13,STF14,STF15,STF17         node 75,76,77,78,79,80,81;
IO10Q1,IO11Q1,IO12Q1,IO13Q1,IO14Q1,IO15Q1,IO17Q1  node 767,768,769,770,771,772,773 ;
IO10Q2,IO11Q2,IO12Q2,IO13Q2,IO14Q2,IO15Q2,IO17Q2  node 127,128,129,130,131,132,133 ;
" INPUT LATCHES
" ===============
IL4,IL5,IL6,IL7,IL8,IL9                           node 813,814,815,816,817,818;
IL10,IL11,IL12,IL13,IL14,IL15,IL17                node 819,820,821,822,823,824,825;
" BURIED LOGIC CELL
" =================
BLC18,BLC19,BLC20,BLC21,BLC22,BLC23               node 178,177,176,175,174,173 ;
declarations "   **** Quadrant II ****
" I/O LOGIC CELL
" ==============
IO18,IO19,IO21,IO22,IO23,IO24                     pin 18,19,21,22,23,24 istype 'buffer,reg_d';
STF18,STF19,STF21,STF22,STF23,STF24               node 82,83,84,85,86,87;
IO18Q1,IO19Q1,IO21Q1,IO22Q1,IO23Q1,IO24Q1         node 774,775,776,777,778,779;
IO18Q2,IO19Q2,IO21Q2,IO22Q2,IO23Q2,IO24Q2         node 134,135,136,137,138,139;
IO25,IO26,IO27,IO28,IO29,IO30,IO31                pin 25,26,27,28,29,30,31 ISTYPE 'BUFFER';
STF25,STF26,STF27,STF28,STF29,STF30,STF31         node 88,89,90,91,92,93,94;
IO25Q1,IO26Q1,IO27Q1,IO28Q1,IO29Q1,IO30Q1,IO31Q1  node 780,781,782,783,784,785,786 ;
IO25Q2,IO26Q2,IO27Q2,IO28Q2,IO29Q2,IO30Q2,IO31Q2  node 140,141,142,143,144,145,146 ;
" INPUT LATCHES
" ===============
IL18,IL19,IL21,IL22,IL23,IL24                     node 826,827,828,829,830,831;
IL25,IL26,IL27,IL28,IL29,IL30,IL31                node 832,833,834,835,836,837,838;
" BURIED LOGIC CELL
" =================
BLC12,BLC13,BLC14,BLC15,BLC16,BLC17               node 184,183,182,181,180,179;
declarations "    **** Quadrant III ****
```

```
" I/O LOGIC CELL
" ==============
IO38,IO39,IO40,IO41,IO42,IO43                       pin 38,39,40,41,42,43;
STF38,STF39,STF40,STF41,STF42,STF43                 node 95,96,97,98,99,100;
IO38Q1,IO39Q1,IO40Q1,IO41Q1,IO42Q1,IO43Q1           node 787,788,789,790,791,792;
IO38Q2,IO39Q2,IO40Q2,IO41Q2,IO42Q2,IO43Q2           node 147,148,149,150,151,152 istype 'reg_t';
IO44,IO45,IO46,IO47,IO48,IO49,IO51                  pin 44,45,46,47,48,49,51;
STF44,STF45,STF46,STF47,STF48,STF49,STF51           node 101,102,103,104,105,106,107;
IO44Q1,IO45Q1,IO46Q1,IO47Q1,IO48Q1,IO49Q1,IO51Q1    node 793,794,795,796,797,798,799;
IO44Q2,IO45Q2,IO46Q2,IO47Q2,IO48Q2,IO49Q2,IO51Q2    node 153,154,155,156,157,158,159;
" INPUT LATCHES
" ===============
IL38,IL39,IL40,IL41,IL42,IL43                       node 839,840,841,842,843,844;
IL44,IL45,IL46,IL47,IL48,IL49,IL51                  node 845,846,847,848,849,850,851;
" BURIED LOGIC CELL
" =================
BLC6,BLC7,BLC8,BLC9,BLC10,BLC11                     node 190,189,188,187,186,185 ISTYPE 'REG_D';
declarations "     **** Quadrant IV ****
" I/O LOGIC CELL
" ==============
IO52,IO53,IO55,IO56,IO57,IO58                       pin 52,53,55,56,57,58 istype 'buffer';
STF52,STF53,STF55,STF56,STF57,STF58                 node 108,109,110,111,112,113;
IO52Q1,IO53Q1,IO55Q1,IO56Q1,IO57Q1,IO58Q1           node 800,801,802,803,804,805;
IO52Q2,IO53Q2,IO55Q2,IO56Q2,IO57Q2,IO58Q2           node 160,161,162,163,164,165;
IO59,IO60,IO61,IO62,IO63,IO64,IO65                  pin 59,60,61,62,63,64,65 istype 'buffer';
STF59,STF60,STF61,STF62,STF63,STF64,STF65           node 114,115,116,117,118,119,120;
IO59Q1,IO60Q1,IO61Q1,IO62Q1,IO63Q1,IO64Q1,IO65Q1    node 806,807,808,809,810,811,812;
IO59Q2,IO60Q2,IO61Q2,IO62Q2,IO63Q2,IO64Q2,IO65Q2    node 166,167,168,169,170,171,172;
" INPUT LATCHES
" ===============
IL52,IL53,IL55,IL56,IL57,IL58                       node 852,853,854,855,856,857;
IL59,IL60,IL61,IL62,IL63,IL64,IL65                  node 858,859,860,861,862,863,864;
" BURIED LOGIC CELL
" =================
BLC0,BLC1,BLC2,BLC3,BLC4,BLC5                       node 196,195,194,193,192,191;
   H,L,C,D,K,U,X,Z = 1,0,.C.,.D.,.K.,.U.,.X.,.Z.;
"   MACRO (INPUT LATCH)
INPUT_LATCH MACRO (IL,IO,QUAD_LE)
{?IL.D  = ?IO;
?IL.LE  = ?QUAD_LE;}
equations
"           INPUT LATCH
"   Each of the 52 I/Os has an input latch. When the quadrant latch enable is high, the
"   pin value is latched.
"   When the quadrant latch clock is low, the latch becomes transparent.

IL4.D   = IO4;    "The .D and the .LE (latch enable) extentions are required to describe a
IL4.LE  = I1;     "latch in. Abel AHDL. The only allowed input to the latch is the IO pin
                  "it associates with.The only allowed .LE input is the quadrant latch
                  "enable. They are Pin 1, 34, 35, and 68.
"  INPUT_LATCH macro is another way to describe the latch.
INPUT_LATCH (IL5,IO5,I1);   "Latch Pin 5
INPUT_LATCH (IL6,IO6,I1);   "Latch Pin 6
IO4.d   = !IO4.fb;  " When .oe is enabled by IO8, IO4 outputs a 1 bit counter.
IO4.oe  = I34;      " When .oe is disabled by !I34, IO4 latches a data bit  from the bus.
IO4.ck  = 1;
IO4.ce  = I2;
IO7 =    IL4;
test_vectors (   "Test the latches...
[I1,I2,I34,IO4]-[IO4,IO7])
[ 0, 0, 1,X  ]  -> [  0,   0 ]; "Transparent
[ 0, C, 1,X  ]  -> [  1,   1 ]; "Transparent
[ U, 0, 0,0  ]  -> [  Z,   0 ]; "Disable .oe and latch 0
```

```
[ 1, C, 1,X ]   -> [  0,   0 ]; "Latched 0
[ 1, C, 1,X ]   -> [  1,   0 ]; "Latched 0
[ D, C, 1,X ]   -> [  0,   0 ]; "Transparent
[ 0, C, 1,X   ]   -> [  1,   1 ]; "Transparent
[ 0, C, 1,X   ]   -> [  0,   0 ]; "Transparent
[ U, 0, 0,1   ]   -> [  Z,   1 ]; "Disable .oe and latch 1
[ 1, 0, 1,X   ]   -> [  0,   1 ]; "Latched 1
[ 1, C, 1,X   ]   -> [  1,   1 ]; "Latched 1
[ 1, C, 1,X   ]   -> [  0,   1 ]; "Latched 1
equations
"  CLOCKING OPTIONS
"  There are different methods of clocking the registers in the ATV5001. The clock is
"  best described as either the AND function of (Quadrant Clock & Clock Product term)
"  or the product term by it self. In the following examples the register can be a name
"  from a pin, Q1, Q2, or Buried Logic Cell. The register can be either .t or.d
"  flip-flop.
IO22.d     =   !IO22.fb;
IO22.ck    =   IO18 & !IO19;  "Note there is no .ce equation defined.This is an asynchronous
IO22.ar    =   I66;           "clocking method  where the quadrant clock  has no effect.

test_vectors (
[IO18,IO19,I66 ]   ->  [ IO22 ])
[   0,    0,1       ]   ->  [     0 ];
[   0,    0,0       ]   ->  [     0 ];
[   C,    0,0       ]   ->  [     1 ];
[   C,    0,0       ]   ->  [     0 ];
equations
IO23.d     =   !IO23.fb;
IO23.ck    =   IO18 & !IO19;  "The clock product term (.ck) is used to gate the quadrant
                              "clock pin.
IO23.ce    =   I32;           "Using quadrant clock pin in a synchronous mode allows higher
                              "clock rate.
IO23.ar    =   I66;           "Pin 32 is the Quadrant 2 clock pin.
test_vectors (
[I32,IO18,IO19,I66  ]   ->     [ IO23])
[   0,   1,0,   1      ]   ->     [ 0];
[   C,   0,0,   0      ]   ->     [ 0]; "Product term blocks quadrant clock
[   C,   1,0,   0      ]   ->     [ 1]; "Product term enables quadrant clock
[   C,   1,0,   0      ]   ->     [ 0];
equations
IO24.d     =   !IO24.fb;
IO24.ck    =   1;      "Note that the .ck is defined to 1. The quadrant clock is the only
                       "clocking element.
IO24.ce    =   I32;    "The .ce is Pin 32 because it's the proper quadrant clock to use for
IO24.ar    =   I66;    "synchronous operation.

test_vectors (
[ I32, I66    ]   ->  [ IO24   ])
[   0,    1   ]   ->  [   0    ];
[   C,    0   ]   ->  [   1    ];
[   C,    0   ]   ->  [   0    ];
[   C,    0   ]   ->  [   1    ];
"   D-TYPE and T-TYPE REGISTERS
"   All the registers in the ATV5001 can be configured as D or T type. The ISTYPE
"   statement after the PIN or NODE definition tells ABEL which type of register is
"   needed.

"   ABC    node   122 istype 'reg_t';
"   XYZ    pin      5 istype 'reg_d';
"   Use the .t extension for ABC and .d extension for XYZ when you write the equations.

equations
" 3 Bit Synchronous Counter using T flip-flops
```

```
IO38Q2.t = 1;                    IO38Q2.ck = 1; IO38Q2.ce = I36;   IO38Q2.ar = I68;
IO39Q2.t = IO38Q2;               IO39Q2.ck = 1; IO39Q2.ce = I36;   IO39Q2.ar = I68;
IO40Q2.t = IO38Q2 & IO39Q2;                     IO40Q2.ck = 1;     IO40Q2.ce = I36;IO40Q2.ar

test_vectors (
[ I36, I68    ]  -> [ IO40Q2,IO39Q2,IO38Q2])
[  0,  1      ]  -> [   0,   0,   0  ];
[  C,  0      ]  -> [   0,   0,   1  ];
[  C,  0      ]  -> [   0,   1,   0  ];
[  C,  0      ]  -> [   0,   1,   1  ];
[  C,  0      ]  -> [   1,   0,   0  ];
[  C,  0      ]  -> [   1,   0,   1  ];
[  C,  0      ]  -> [   1,   1,   0  ];
[  C,  0      ]  -> [   1,   1,   1  ];
[  C,  0      ]  -> [   0,   0,   0  ];

equations
" 3 Bit Synchronous Counter using D flip-flops
BLC6.d = !BLC6;                  BLC6.ck = 1;       BLC6.ce = I36;        BLC6.ar = I68;
BLC7.d = BLC6 $ BLC7;            BLC7.ck = 1;       BLC7.ce = I36;        BLC7.ar = I68;
BLC8.d = BLC8 $ (BLC6 & BLC7);   BLC8.ck = 1;       BLC8.ce = I36;        BLC8.ar = I68;

test_vectors (
[ I36, I68    ]  -> [BLC8,BLC7,BLC6])
[  0,  1      ]  -> [  0,   0,   0  ];
[  C,  0      ]  -> [  0,   0,   1  ];
[  C,  0      ]  -> [  0,   1,   0  ];
[  C,  0      ]  -> [  0,   1,   1  ];
[  C,  0      ]  -> [  1,   0,   0  ];
[  C,  0      ]  -> [  1,   0,   1  ];
[  C,  0      ]  -> [  1,   1,   0  ];
[  C,  0      ]  -> [  1,   1,   1  ];
[  C,  0      ]  -> [  0,   0,   0  ];
"        UNIVERSAL AND REGIONAL PRODUCT TERMS
"   A Regional product term has inputs from all the feedbacks of its quadrant Buried
"   Logic Cells, Q1's, Q2's,and the dedicated Inputs. In addition, a Universal product
"   term has the Pin/B Sum Term Feedbacks.
"        OUTPUTS and FEEDBACKS
"   Combinatorial:
"   Use IO pin name to define the equation. The sum terms may combine to allow 5, 9, or
"   13 product terms depending on the need of the reduced equation. If the reduced
"   equation requires 5 or less product terms, it leaves Q1 and Q2 each with 4 product
"   terms to use.
"        IO52               =    up to 5 PRODUCT TERMS (3UNIVERSAL)
"        IO52.oe            =    1 PRODUCT TERM (REGIONAL)
"        IO52Q1.(d or t) =       up to 4 PRODUCT TERMS (2 UNIVERSAL)
"        IO52Q1.ck          =    1 PRODUCT TERM (1 UNIVERSAL)
"        IO52Q1.ce          =    QUADRANT CLOCK
"        IO52Q1.ar          =    1 PRODUCT TERM  (REGIONAL)
"        IO52Q1.ap          =    1 PRODUCT TERM  (REGIONAL)
"        IO52Q2.(d or t) =       up to 4 PRODUCT TERMS (3 UNIVERSAL)
"        IO52Q2.ck          =    1 PRODUCT TERM (1 UNIVERSAL)
"        IO52Q2.ce          =    QUADRANT CLOCK
"        IO52Q2.ar          =    1 PRODUCT TERM (REGIONAL)
"        IO52Q2.ap          =    1 PRODUCT TERM (REGIONAL)
"   ALLOWED FEEDBACKS:             SOURCE:
"   IO52                  --       Pin feedback, after the buffer/inverter.
"   IO52Q1,IO52Q1.FB      --       Q1 register feedback.
"   IO52Q2,IO52Q2.FB      --       Q2 register feedback.
"   Combinatorial:
"   If the reduced equation requires 6 to 9 product terms, it leaves Q2 with 4 product
"   terms to use. Q1 will feedback the A Sum Term portion of the output logic.
```

```
"       IO52              =    up to 9 PRODUCT TERMS (5 UNIVERSAL)
"       IO52.oe           =    1 PRODUCT TERM (REGIONAL)
"       IO52Q2.(d or t)   =    up to 4 PRODUCT TERMS (3 UNIVERSAL)
"       IO52Q2.ck         =    1 PRODUCT TERM (1 UNIVERSAL)
"       IO52Q2.ce         =    QUADRANT CLOCK
"       IO52Q2.ar         =    1 PRODUCT TERM (REGIONAL)
"       IO52Q2.ap         =    1 PRODUCT TERM (REGIONAL)
"  ALLOWED FEEDBACKS:             SOURCE:
"  IO52                --      Pin feedback, after the buffer/inverter.
"  IO52Q2,IO52Q2.FB    --      Q2 register feedback.
"  Combinatorial:
"  If the reduced equation requires more than 9 product terms, it leaves no product
"  terms for Q1 and Q2. Q1 feeds back A Sum Term and Q2 feeds back C Sum Term.
"       IO52              =    up to 13 PRODUCT TERMS (8 UNIVERSAL)
"       IO52.oe           =    1 PRODUCT TERM (REGIONAL)
"  ALLOWED FEEDBACKS:             SOURCE:
"  IO52                --      Pin feedback, after the buffer/inverter.
"  Registered:
"  Use IO pin name to define the equation. The sum terms may combine to allow 4, 9, or
"  13 product terms depending on the need of the reduced equation. If the reduced
"  equation requires 4 or less product terms, you may define a 5 product term equation
"  for the STF (Sum Term Feedback) and define a 4 product term equation for Q2.
"       IO52.(d or t)     =    up to 4 PRODUCT TERMS (2 UNIVERSAL)
"       IO52.oe           =    1 PRODUCT TERM (REGIONAL)
"       IO52.ck           =    1 PRODUCT TERM (1 UNIVERSAL)
"       IO52.ce           =    QUADRANT CLOCK
"       IO52.ar           =    1 PRODUCT TERM (REGIONAL)
"       IO52.ap           =    1 PRODUCT TERM (REGIONAL)
"       STF52             =    up to 5 PRODUCT TERMS (3 UNIVERSAL)
"       IO52Q2.(d or t)   =    up to 4 PRODUCT TERMS (3 UNIVERSAL)
"       IO52Q2.ck         =    1 PRODUCT TERM (1 UNIVERSAL)
"       IO52Q2.ce         =    QUADRANT CLOCK
"       IO52Q2.ar         =    1 PRODUCT TERM (REGIONAL)
"       IO52Q2.ap         =    1 PRODUCT TERM (REGIONAL)

"  ALLOWED FEEDBACKS:             SOURCE:
"  IO52.FB,IO52.Q      --      Register feedback prior to buffer/inverter.
"  IO52Q2              --      Q2 register feedback.
"  STF52               --      Sum Term Feedback for logic expansion.
"  Registered:
"  If the reduced equation requires 9 to 5 product terms, you may write a 4 product
"  term equation for Q2.
"       IO52.(d or t)     =    up to 9 PRODUCT TERMS (5 UNIVERSAL)
"       IO52.oe           =    1 PRODUCT TERM (REGIONAL)
"       IO52.ck           =    1 PRODUCT TERM (1 UNIVERSAL)
"       IO52.ce           =    QUADRANT CLOCK
"       IO52.ar           =    1 PRODUCT TERM (REGIONAL)
"       IO52.ap           =    1 PRODUCT TERM (REGIONAL)
"       IO52Q2.(d or t)   =    up to 4 PRODUCT TERMS (3 UNIVERSAL)
"       IO52Q2.ck         =    1 PRODUCT TERM (1 UNIVERSAL)
"       IO52Q2.ce         =    QUADRANT CLOCK
"       IO52Q2.ar         =    1 PRODUCT TERM (REGIONAL)
"       IO52Q2.ap         =    1 PRODUCT TERM (REGIONAL)
"  ALLOWED FEEDBACKS:             SOURCE:
"  IO52                --      Pin feedback, after the buffer/inverter.
"  IO52.FB             --      Register feedback prior to buffer/inverter.
"  IO52Q2              --      Q2 register feedback.
"  Registered:
"  If the reduced equation requires more than 9 product terms, it leaves zero product
terms for Q1 and Q2.
"       IO52.(d or t)     =    up to 13 PRODUCT TERMS (8 UNIVERSAL)
"       IO52.oe           =    1 PRODUCT TERM (REGIONAL)
"       IO52.ck           =    1 PRODUCT TERM (1 UNIVERSAL)
```

```
"       IO52.ce         =  QUADRANT CLOCK
"       IO52.ar         =  1 PRODUCT TERM (REGIONAL)
"       IO52.ap         =  1 PRODUCT TERM (REGIONAL)
"  ALLOWED FEEDBACKS:       SOURCE:
"  IO52                 --  Pin feedback, after the buffer/inverter.
"  IO52.FB              --  Register feedback prior to buffer/inverter.
"       AP, AR, and CK
"  Each of the 128 registers has its own .AP (Asynchronous Preset), .AR (Asynchronous
"  Reset) and .CK. AlL CKs are Universal product terms. The ARs and APs are Regional
"  product terms.
"  NOTE: AP and AR should never be active at the same time.

equations
IO10.d     =  !IO10.fb;
IO10.ck    =  I08;
IO10.ar    =  I66;
IO10.ap    =  I35;

test_vectors (
[ I08, I66, I35 ] -> [ IO10 ])
[  0,    1, 0   ] -> [    0 ]; "AR
[  0,    0, 0   ] -> [    0 ];
[  C,    0, 0   ] -> [    1 ];
[  C,    0, 0   ] -> [    0 ];
[  0,    0, 1   ] -> [    1 ]; "AP
[  0,    1, 0   ] -> [    0 ]; "AR
"         BURIED LOGIC CELLS
"  Buried Logic Cell can be configured as a register feedback like Q2. It can also be
"  configured as  combinatorial feedback to accommodate for logic expansion.

equations
BLC17.t    =  1; "Registered
BLC17.ck   =  I09;
BLC17.ar   =  I66;
BLC14      =  I1; "Combinatiorial
IO25       =  BLC17;
IO26       =  BLC14;

test_vectors (
[I09, I66,    I1 ]  -> [ BLC17,BLC14,IO25,IO26])
[   0,   1, 0    ]  -> [    0,  0,   0,   0  ];
[   C,   0, 0    ]  -> [    1,  0,   1,   0  ];
[   0,   0,    1 ]  -> [    1,  1,   1,   1  ];
[   C,   0, 1    ]  -> [    0,  1,   0,   1  ];
[   C,   0,    0 ]  -> [    1,  0,   1,   0 ];
END;
```

Example Cupl™ Description File

```
Name      V5100CU;
Partno    NA;
Date      5/20/92;
Revision  01;
Designer  Joe Yu;
Company   Atmel Corporation;
Assembly  NA;
Location  U1;
Device    V5100;
Format    j;
/* Compiled with Cupl 386 version */
/* The IOs, registers, inputs, and combinatorial sum terms feedbacks are named with
/* the following prefixes for clarity:

*  Prefix
 * I      -  Quadrant Clocks, Latch Enables.
 * IO     -  IO pins
 * BLC    -  Buried Logic Cells
 * STF    -  Sum Term Feedbacks
 * Valid CUPL HDL identifiers can be used in place of them. */

pin 1   = I1;       /* Quadrant 1              Latch Enable/Input      */
pin 2   = I2;       /* Quadrant 1              Synchronous Register Clock/Input*/
pin 32  = I32;      /* Quadrant 2              Synchronous Register Clock/Input*/
pin 34  = I34;      /* Quadrant 2              Latch Enable/Input      */
pin 35  = I35;      /* Quadrant 3              Latch Enable/Input      */
pin 36  = I36;      /* Quadrant 3              Synchronous Register Clock/Input*/
pin 66  = I66;      /* Quadrant 4              Synchronous Register Clock/Input*/
pin 68  = I68;      /* Quadrant 4              Latch Enable/Input      */
          /**** Quadrant I ****/
/* I/O LOGIC CELL  */
/* ============== */
pin       [4..9]          = [IO4..9];
pinnode   [209..204]      = [STF4..9];
pinnode   [157..152]      = [IO4Q1,IO5Q1,IO6Q1,IO7Q1,IO8Q1,IO9Q1];
pinnode   [105..100]      = [IO4Q2,IO5Q2,IO6Q2,IO7Q2,IO8Q2,IO9Q2];
pin       [10..15,17]     = [IO10..15,IO17];
pinnode   [203..197]      = [STF10..15,STF17];
pinnode   [151..145]      = [IO10Q1,IO11Q1,IO12Q1,IO13Q1,IO14Q1,IO15Q1,IO17Q1];
pinnode   [99..93]        = [IO10Q2,IO11Q2,IO12Q2,IO13Q2,IO14Q2,IO15Q2,IO17Q2];
/* BURIED LOGIC CELL    */
/* ================= */
pinnode   [69..74]        = [BLC18..23];
          /**** Quadrant II ****/
/* I/O LOGIC CELL */
/* ============== */
pin       [18,19,21..24]  = [IO18,IO19,IO21..24];
pinnode   [210..215]      = [STF18,STF19,STF21..24];
pinnode   [158..163]      = [IO18Q1,IO19Q1,IO21Q1,IO22Q1,IO23Q1,IO24Q1];
pinnode   [106..111]      = [IO18Q2,IO19Q2,IO21Q2,IO22Q2,IO23Q2,IO24Q2];
pin       [25..31]        = [IO25..31];
pinnode   [216..222]      = [STF25..31];
pinnode   [164..170]      = [IO25Q1,IO26Q1,IO27Q1,IO28Q1,IO29Q1,IO30Q1,IO31Q1];
pinnode   [112..118]      = [IO25Q2,IO26Q2,IO27Q2,IO28Q2,IO29Q2,IO30Q2,IO31Q2];
/* BURIED LOGIC CELL */
/* ================= */
pinnode   [80..75]        = [BLC12..17];
          /**** Quadrant III ****/
/* I/O LOGIC CELL  */
/* ============== */
```

```
pin         [38..43]           = [IO38..43];
pinnode     [235..230]         = [STF38..43];
pinnode     [183..178]         = [IO38Q1,IO39Q1,IO40Q1,IO41Q1,IO42Q1,IO43Q1];
pinnode     [131..126]         = [IO38Q2,IO39Q2,IO40Q2,IO41Q2,IO42Q2,IO43Q2];
pin         [44..49,51]        = [IO44..49,IO51];
pinnode     [229..223]         = [STF44..49,STF51];
pinnode     [177..171]         = [IO44Q1,IO45Q1,IO46Q1,IO47Q1,IO48Q1,IO49Q1,IO51Q1];
pinnode     [125..119]         = [IO44Q2,IO45Q2,IO46Q2,IO47Q2,IO48Q2,IO49Q2,IO51Q2];
/* BURIED LOGIC CELL    */
/* =================    */
pinnode     [81..86]           = [BLC6..11];
            /**** Quadrant IV ****/
/* I/O LOGIC CELL    */
/* ==============    */
pin         [52,53,55..58]     = [IO52,IO53,IO55..58];
pinnode     [236..241]         = [STF52,STF53,STF55..58];
pinnode     [184..189]         = [IO52Q1,IO53Q1,IO55Q1,IO56Q1,IO57Q1,IO58Q1];
pinnode     [132..137]         = [IO52Q2,IO53Q2,IO55Q2,IO56Q2,IO57Q2,IO58Q2];
pin         [59..65]           = [IO59..65];
pinnode     [242..248]         = [STF59..65];
pinnode     [190..196]         = [IO59Q1,IO60Q1,IO61Q1,IO62Q1,IO63Q1,IO64Q1,IO65Q1];
pinnode     [138..144]         = [IO59Q2,IO60Q2,IO61Q2,IO62Q2,IO63Q2,IO64Q2,IO65Q2];
/* BURIED LOGIC CELL    */
/* =================    */
pinnode     [92..87]           = [BLC0..5];
/*          INPUT LATCH
   Each of the 52 I/Os has an input latch. When the quadrant latch enable is high, the
   pin value is latched.  When the quadrant latch clock is low, the latch becomes
   transparent.    */
IO4.d       =       !IO4;
IO4.oe      =       I34;
IO4.ce      =       'B'1;
IO7         =       IO4.IOL;    /* When the latched input is needed in the design, i.e.,  */
                                /*IO4.IOL, IO4 is  configured as  having a latched input. */
/*          CLOCKING OPTIONS
   There are different methods of clocking the registers in the ATV5100. The clock is
   best described as either the AND function of (Quadrant Clock & Clock Product term) or
   the product term by it self. In the following examples the register can be a name
   from a pin, Q1, Q2, or Buried Logic Cell. The register can be either  .t or .d
   flip-flop.    */
IO22.d      =       !IO22;
IO22.ck     =       IO18 & !IO19; /* Note there is no .ce equation defined. This is an    */
IO22.ar     =       I66;          /* asynchronous clocking method where the quadrant      */
                                  /* clock has no effect.                                  */
IO23.d      =       !IO23;
IO23.ce     =       IO18 & !IO19; /* The clock product term is used to gate the quadrant */
IO23.ar     =       I66;          /* clock pin.Using quadrant clock pin in a synchronous */
                                  /* mode allows higher clock rate. Pin 32 is the        */
                                  /* Quadrant 2 clock pin.                               */

IO24.d  = !IO24;
IO24.ce    =   'B'1;              /* Note that the .ce is defined to 1. The quadrant     */
IO24.ar    =   I66;               /* clock is the only clocking element. The clock is    */
                                  /* Pin 32 because it's the proper quadrant clock to    */
                                  /* use for synchronous operation.                      */
/*          D-TYPE and T-TYPE REGISTERS
   All the registers in the ATV5100 can be configured as D or T type. Use the .d
   extension for D type register and .t extension for T type register. */
/* 3 Bit Synchronous Counter using T flip-flops */
IO38Q2.t = 'B'1;                  IO38Q2.ce = 'B'1;  IO38Q2.ar = I68;
IO39Q2.t = IO38Q2;                IO39Q2.ce = 'B'1;  IO39Q2.ar = I68;
IO40Q2.t = IO38Q2 & IO39Q2;       IO40Q2.ce = 'B'1;  IO40Q2.ar = I68;
/* 3 Bit Synchronous Counter using D flip-flops */
```

```
BLC6.d = !BLC6;                    BLC6.ce = 'B'1;     BLC6.ar = I68;
BLC7.d = BLC6 $ BLC7;              BLC7.ce = 'B'1;     BLC7.ar = I68;
BLC8.d = BLC8 $ (BLC6 & BLC7);     BLC8.ce = 'B'1;     BLC8.ar = I68;
/*      UNIVERSAL AND REGIONAL PRODUCT TERMS
   A Regional product term has inputs from all the feedbacks of its quadrant Buried
   Logic Cells, Q1's, Q2's, and the dedicated Inputs. In addition, a Universal product
   term has the Pin/B Sum Term Feedbacks.        */
/*       OUTPUTS and FEEDBACKS
   Combinatorial:
   Use IO pin name to define the equation. The sum terms may combine to allow 5, 9, or
   13 product terms  depending on the need of the reduced equation. If the reduced
   equation requires 5 or less product terms,it leaves Q1 and Q2 each with 4 product
   terms to use.
        IO52                 = up to 5 PRODUCT TERMS (3 UNIVERSAL)
        IO52.oe              = 1 PRODUCT TERM (REGIONAL)
        IO52Q1.(d or t)      = up to 4 PRODUCT TERMS (2 UNIVERSAL)
        IO52Q1.(ck or ce)    = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52Q1.ar            = 1 PRODUCT TERM (REGIONAL)
        IO52Q1.ap            = 1 PRODUCT TERM (REGIONAL)
        IO52Q2.(d or t)      = up to 4 PRODUCT TERMS (3 UNIVERSAL)
        IO52Q2.(ck or ce)    = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52Q2.ar            = 1 PRODUCT TERM (REGIONAL)
        IO52Q2.ap            = 1 PRODUCT TERM (REGIONAL)
   ALLOWED FEEDBACKS:        SOURCE:
   IO52,IO52.IO       --     Pin feedback, after the buffer/inverter.
   IO52Q1             --     Q1 register feedback.
   IO52Q2             --     Q2 register feedback.        */
/* Combinatorial:
   If the reduced equation requires 6 to 9 product terms, it leaves Q2 with 4 product
   terms to use. Q1 will feedback the A Sum Term portion of the output logic.
        IO52                 = up to 9 PRODUCT TERMS (5 UNIVERSAL)
        IO52.oe              = 1 PRODUCT TERM (REGIONAL)
        IO52Q2.(d or t)      = up to 4 PRODUCT TERMS (3 UNIVERSAL)
        IO52Q2.(ck or ce)    = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52Q2.ar            = 1 PRODUCT TERM (REGIONAL)
        IO52Q2.ap            = 1 PRODUCT TERM (REGIONAL)
   ALLOWED FEEDBACKS:        SOURCE:
   IO52,IO52.IO       --     Pin feedback, after the buffer/inverter.
   IO52Q2             --     Q2 register feedback.        */
/* Combinatorial:
   If the reduced equation requires more than 9 product terms, it leaves no product
   terms for Q1 and Q2. Q1 feeds back A. Sum Term and Q2 feeds back C Sum Term.
        IO52        =  up to 13 PRODUCT TERMS (8UNIVERSAL)
        IO52.oe     =  1 PRODUCT TERM (REGIONAL)
   ALLOWED FEEDBACKS:        SOURCE:
   IO52,IO52.IO       --     Pin feedback, after the buffer/inverter. */
/* Registered:
   Use IO pin name to define the equation. The sum terms may combine to allow 4, 9, or
   13 product terms depending on the  need of the reduced equation. If the reduced
   equation requires 4 or less product terms, you may define a 5 product term equation
   for the STF (Sum Term Feedback) and define a 4 product term equation for Q2.
        IO52.(d or t)        = up to 4 PRODUCT TERMS (3 UNIVERSAL)
        IO52.oe              = 1 PRODUCT TERM (REGIONAL)
        IO52.(ck or ce)      = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52.ar              = 1 PRODUCT TERM (REGIONAL)
        IO52.ap              = 1 PRODUCT TERM (REGIONAL)

        STF52                = up to 5 PRODUCT TERMS (3 UNIVERSAL)
        IO52Q2.(d or t)      = up to 4 PRODUCT TERMS (3 UNIVERSAL)
        IO52Q2.(ck or ce)    = 1 PRODUCT TERM (1 UNIVERSAL)
        IO52Q2.ar            = 1 PRODUCT TERM (REGIONAL)
        IO52Q2.ap            = 1 PRODUCT TERM (REGIONAL)
```

```
    ALLOWED FEEDBACKS:          SOURCE:
    IO52                --      Register feedback prior to buffer/inverter.
    IO52Q2              --      Q2 register feedback.
    STF52               --      Sum Term Feedback for logic expansion.  */
/*  Registered:
    If the reduced equation requires 9 to 5 product terms, you may write a 4 product term
    equation for Q2.
         IO52.(d or t)         = up to 9 PRODUCT TERMS (5 UNIVERSAL)
         IO52.oe               = 1 PRODUCT TERM (REGIONAL)
         IO52.(ck or ce)       = 1 PRODUCT TERM (1 UNIVERSAL)
         IO52.ar               = 1 PRODUCT TERM (REGIONAL)
         IO52.ap               = 1 PRODUCT TERM (REGIONAL)
         IO52Q2.(d or t)       = up to 4 PRODUCT TERMS (3 UNIVERSAL)
         IO52Q2.(ck or ce)     = 1 PRODUCT TERM (1 UNIVERSAL)
         IO52Q2.ar             = 1 PRODUCT TERM (REGIONAL)
         IO52Q2.ap             = 1 PRODUCT TERM (REGIONAL)
    ALLOWED FEEDBACKS:          SOURCE:
    IO52.IO             --      Pin feedback, after the buffer/inverter.
    IO52                --      Register feedback prior to buffer/inverter.
    IO52Q2              --      Q2 register feedback. */
/*  Registered:
    If the reduced equation requires more than 9 product terms, it leaves zero product
    terms for Q1 and Q2.
         IO52.(d or t)         = up to 13 PRODUCT TERMS (8 UNIVERSAL)
         IO52.oe               = 1 PRODUCT TERM (REGIONAL)
         IO52.(ck or ce)       = 1 PRODUCT TERM (1 UNIVERSAL)
         IO52.ar               = 1 PRODUCT TERM (REGIONAL)
         IO52.ap               = 1 PRODUCT TERM (REGIONAL)
    ALLOWED FEEDBACKS:          SOURCE:
    IO52.IO             --      Pin feedback, after the buffer/inverter.
    IO52                --      Register feedback prior to buffer/inverter.*/
/*          AP, AR, and CK
    Each of the 128 registers has its own .AP (Asynchronous Preset), .AR (Asynchronous
    Reset) and .CK or .CE. All CKs are Universal product terms. The ARs and APs are
    Regional product terms.
    NOTE:  AP and AR should never be active at the same time.           */
IO10.d  = !IO10;
IO10.ck        =       IO8;
IO10.ar        =       I66;
IO10.ap        =       I35;
/*      BURIED LOGIC CELLS
    Buried Logic Cell can be configured as a register feedback like Q2. It can also be
    configured as combinatorial feedback to accommodate for logic expansion. */
BLC17.t        =       'B'1;    /*Registered */
BLC17.ck       =       IO9;
BLC17.ar       =       I66;
BLC14          =       I1;      /*Combinatiorial */
IO25           =       BLC17;
IO26           =       BLC14;
END;
```

Using Programmable Logic Devices

UV Erasable Programmable Logic Device

Application Note

Introduction

This application note covers three areas:

- Where and *why* do I use Programmable Logic Devices (*PLDs*)?
- *How* do I use PLDs?
- Software and hardware *support* for Atmel PLDs.

Where Do I Use PLDs?

Any digital logic design can be done using PLDs. If you normally begin your design by:

- Using AND and OR functions
- Thinking of 7400 series components
- Using truth tables, or
- State diagrams

you are already on the path to using PLDs.

Designing a microprocessor based system, with memory and I/O? How about all that "glue" logic you use to interface with the bus, provide chip selects, and any unusual signals required by special chips? Most of these functions are currently done with 7400 series TTL. *How about using a PLD instead?*

Designing a stand-alone PC board which uses a state machine to control multiple output signals? Using latches to synchronize signals? Using counters to divide down master clock frequencies? Converting parallel-to-serial and back again? All of these functions fit easily in modern PLDs. *Most anything found in your TTL Databook can be replaced with your own, PERSONALIZED, programmable logic device.*

PLD Applications

- **Glue Logic**
- **State Machines**
- **Synchronization**
- **Decoders**
- **Counters**
- **Bus Interfaces**
- **Parallel-to-Serial**
- **Serial-to-Parallel**
- **Subsystems**
- **and Many Others**

Why PLDs?

Maybe you have already heard all the wonderful reasons for using PLDs. Well, they're true! First, let's review some of the more important ones:

- **Increased Integration.** You can reduce the package count of your designs while simultaneously increasing the features offered by your product.
- **Lower Power.** CMOS and fewer packages combine to reduce power consumption.
- **Improved Reliability.** Lower power plus fewer interconnections and packages translate into greatly improved system reliability.
- **Lower Cost.** PLDs reduce inventory costs, too.
- **Easier To Use!** Yes, believe it or not, once you get past the initial learning period, PLDs are easier to use than discrete logic functions.
- **Easier to Change.** Oops! Need to make a change? You won't need "blue wire" when you use a PLD— all changes are internal, and can be done quickly. ECNs are a snap— and system reliability is maintained!

Let's Get Started!

Figure 1 describes the PLD design process. After having read the first part of this application note, you now have the perfect application for a PLD, right? So here you go!

How do you translate your idea into a working prototype? *First, you need a computer with an editor of some kind.* If you have a workstation with a schematic editor, you may input your design using familiar logic blocks. Otherwise, a line or full screen text editor, used in the non-document mode will do. An example of an Abel™ text file is on the next page.

Next, turn the **logic compiler** loose on your design. First it will *check for typographical errors* and any inconsistencies in your specification. Most compilers then attempt to *reduce your logic* using standard logic reduction theory. Then, a **simulator** will check the test vectors you input, comparing your logic description against the predicted responses. This is an excellent way to verify your design. Check with the appropriate software manuals for more information.

At the end of the compilation process, a *JEDEC* file is output. This file is a standard format accepted by most programming hardware. Next *download this file* to your chosen programmer.

At this point you are ready to "build" your **prototype.** Make sure the programmer has the correct information to program the device you have chosen (an Atmel PLD, of course), plug in your device, and go! Most programmers will even functionally test your prototype for you if you include test vectors in your JEDEC file.

Take your configured PLD, and *plug it into your system.* If you find any errors, just use your editor to make the necessary changes, and repeat the process. It's easy!

Example Design

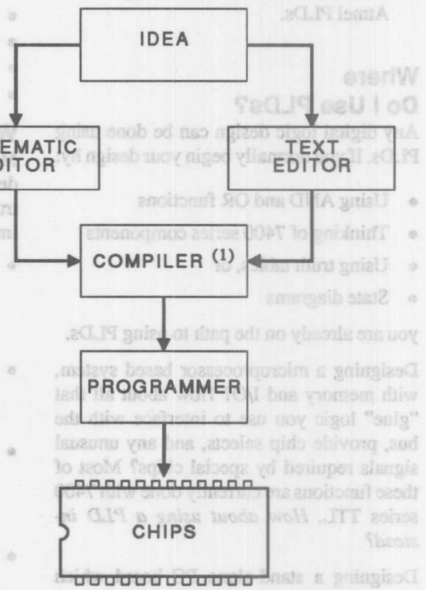
The following design is a simple example using Abel™ to process the logic description file and an AT22V10 as the target device. The equations are on the next page, and are a direct reproduction of the actual Abel input file.

Each of the three allowable input formats are shown. A truth table is used to describe a simple 2-to-4 decoder, as is often used to decode chip selects in a microprocessor system. Next, the state machine format is used to describe a divide-by-4 counter. And finally, Boolean equations are used to describe some random logic.

Note the test vectors used to test the device. The "c" nomenclature means that this pin has a low to high to low series of transitions for this vector. Each time this happens, the counter should increment. Also note that the counter starts in the reset condition, which is both outputs "1" for an active low output.

Now you're ready to go - Have fun!

Figure 1. PLD Design Process



Note: 1. Examples of compilers are Abel™, Cupl™, and LOG/iC. Each of these products contains modules which allow simulation of your design. They also minimize your logic equations, which gives you flexibility in describing your design.

Example Abel™ Description File

```
module X3;
title 'Example using 22V10 - KHG 1/6/88';
X310 device 'P22V10';
"
Clk,A12,A13      pin    1,2,3;
CE0,CE1,CE2,CE3 pin    20,21,22,23;
Q1,Q2,CarOut     pin    17,18,14;
CarEn,A,B,C,D    pin    6,7,8,9,10;
Out1,Out2        pin    15,16;
"
X,Z,c     =  .X. , .Z. , .C.;
"
"Counter States
State1    =  ^b00;    State2    =  ^b01;
State3    =  ^b10;    State4    =  ^b11;
"
"The following truth table defines the 2 to 4 decoder, which decodes
"  A13 and A12 into CE0, CE1, CE2, and CE3.
truth_table ([A13,A12] -> [CE0,CE1,CE2,CE3])
[ 0, 0 ]  ->  [ 0, 1, 1, 1 ];
[ 0, 1 ]  ->  [ 1, 0, 1, 1 ];
[ 1, 0 ]  ->  [ 1, 1, 0, 1 ];
[ 1, 1 ]  ->  [ 1, 1, 1, 0 ];
"The following state description defines the divide by 4 counter
state_diagram [Q2,Q1]
State State1:   GOTO   State2;
State State2:   GOTO   State3;
State State3:   GOTO   State4;
State State4:   GOTO   State1;
"  The following equations are general in nature to illustrate Boolean input
"  format.  The CarOut equation uses state 4 above to produce a carry.
Equations
CarOut=  Q2  &  Q1  &  CarEn;     "& = AND
Out1  =  A &  B +  C &  D;  "+ = OR, AND takes precedence
Out2  =  A &  C +  B &  D;
"The following are the appropriate test vectors
test_vectors
"
([Clk, CarEn, A13,A12, A, B, C, D]- > [CE0,CE1,CE2,CE3,Q2,Q1,CarOut,Out1,Out2]);
[ 0, 0, 0, 0, 0, 0, 0, 0 ] - > [ 0, 1, 1, 1, 1, 1, 0, 0, 0 ];
[ c, 0, 0, 1, 1, 1, 0, 0 ] - > [ 1, 0, 1, 1, 0, 0, 0, 1, 0 ];
[ c, 0, 1, 0, 1, 0, 1, 0 ] - > [ 1, 1, 0, 1, 0, 1, 0, 0, 1 ];
[ c, 0, 1, 1, 0, 0, 1, 1 ] - > [ 1, 1, 1, 0, 1, 0, 0, 1, 0 ];
[ c, 0, 0, 0, 0, 1, 0, 1 ] - > [ 0, 1, 1, 1, 1, 1, 0, 0, 1 ];
[ 0, 1, 0, 1, 1, 1, 1, 1 ] - > [ 1, 0, 1, 1, 1, 1, 1, 1, 1 ];

end X3;
```

5

Selecting Decoupling Capacitors for Atmel's PLDs

UV Erasable Programmable Logic Device

Application Note

Introduction

This application note provides a summary of information needed when selecting decoupling capacitors for Atmel Programmable Logic Devices. A 0.22-µF, multi-layer ceramic or plastic dielectric capacitor is recommended for such use. Either surface-mount (SMD) or radial-leaded devices should be used. Because of their high parasitic resistance and/or inductance, tantalum, aluminum electrolytic, and axially leaded capacitors are not recommended.

When is a Capacitor Not a Capacitor

Unfortunately, capacitors are not the perfect charge storage devices we would like them to be. Their lead wires and internal construction create parasitic resistance and inductance in series with the capacitance. These parasitics are usually referred to as ESR (equivalent series resistance) and ESL (equivalent series inductance), respectively. As will be shown, these parasitics can seriously reduce the ability of many types of capacitors to decouple supply noise in high-speed systems. Table 1 gives typical ESR and ESL values for various types of capacitors.

As shown, ESR values range from 0.01 ohm to as high as 9 ohms. ESL varies from 2 nH for typical surface mount devices to 20 nH for electrolytic capacitors. These numbers are typical values, taken from data from several manufacturers. As expected, there is some variation between manufacturers. Also, worst case specification values will be significantly higher, especially for ESR values.

How ESR and ESL can Affect High Speed Operation

The effects of these parasitics may be best illustrated by a simple example. Consider the case of a 22V10L. In the standby mode, Icc current is typically only 5 mA. When an input switches, Icc may temporarily go as high as 100 mA. This increase in current draws charge from the local decoupling capacitor. This capacitor current will create voltage drops across the ESR and ESL parasitic elements. To see how these voltage drops can cause problems in a system, look at a typical decoupling application.

In this example the design goal of the capacitor is to keep local supply noise below 0.2 volts, a reasonable expectation. This immediately sets an upper limit on ESR of 2 ohms.

$ESR_{max} = V_{noise} / I_{max}$

Imax= Highest Expected Capacitor Current

The upper limit on ESL is determined by how quickly the capacitor's current must change, as well as how much supply noise will be tolerated during that change. For high-speed logic devices, Icc must be able to switch from standby to active levels within 2 to 3 nanoseconds.

$ESL_{max} = V_{noise} \cdot I_{max} / \Delta t;$

Δt = Time allowed for capacitor current to switch

In this example, an upper limit on ESL of 4 to 6 nH is set.

Consider what can happen if these limits are exceeded. If an axially leaded multi-layer ceramic capacitor with ESR of 0.15 ohm is used, the resistance drop in our application

5

will not be significant (100 mA X .15 Ohm = 15 mV). However, the inductance will not allow the current to reach 100 mA in 6 nanoseconds. This can slow the logic device switching by several nanoseconds.

What Types to Use: Multi-layered Ceramic and Plastic Dielectrics

From this example, it is apparent that the parasitic elements on capacitors can easily limit their decoupling ability. Therefore, users of high-speed logic need to pick their capacitors with care. The data in Table 1 shows that the best bets are surface-mount, multi-layered ceramic (MLC) or plastic dielectrics. Of the leaded devices, only radial types are recommended.

Within the MLCs, there are different classes of dielectric. Class I has the best characteristics, but its small dielectric constant makes it impractical for decoupling values. Class II is highly recommended, as it has good temperature stability (percent variation -55°C to 125°C) and aging characteristics (10 percent in ten years). Class III, on the other hand, drops to less than 50 percent of its rated capacitance at 85°C, and to only 25 percent at -55°C. Class III dielectric also loses 20 percent of its rated value in ten years. Therefore, Class III MLCs are only recommended for applications where temperature excursions are minimal.

Plastic dielectric capacitors in general offer performance as good as Class II MLCs. Among the dielectrics available today are polypropylene, polyester, polycarbonate, polystyrene and teflon. Capacitance variation with temperature depends on the particular material, but is generally less than ±20 percent from -55°C to 125°C. Aging is minimal, usually less than 2 percent in 10 years. Unfortunately, not many manufacturers make surface-mount plastic dielectric capacitors. That should change soon, as surface-mount technology advances and becomes more common.

When using radial leaded cases, be sure to minimize lead lengths, as ESL increases quickly with longer leads. For example, if a capacitor has 6 nH of inductance with 2 mm leads, extending leads to 5 mm will increase ESL to 10 nH.

What Types Not to Use: Aluminum Electrolytic, Tantalum, and Anything Axial

The design example above together with the numbers given in Table 1 show that some types are not suitable at all for decoupling high-speed devices. Specifically, the high inductance of axially leaded capacitors puts them on the "don't use" list. Also, tantalum and aluminum electrolytic devices are generally not recommended, as they have high ESR and/or ESL, even in radial and surface-mount configurations.

In Any Case, Know Your ESL and ESR

ESR data is often found in catalogs. However, this will normally be only low-frequency data, and ESR is frequency dependent (dropping at higher f). ESL data is not usually given in catalogs. The best thing to do is get Z versus frequency data from the manufacturer. From such a graph (with frequency up to at least 10 MHz), you can extract high frequency ESR and ESL.

How Much Capacitance Do You Need

For decoupling Atmel's PLDs a 0.22-µF capacitor is recommended. In many cases, this will be overkill. However, determining how much less you could get by with for a particular application is dependent upon several factors. The number of PC board supply planes, the board's dielectric thickness and dielectric constant, the value (*and* ESR *and* ESL!) of power entry decoupling capacitors, among other things, will determine just how much is really needed. The best bet is to use a good 0.22-µF and be safe. Besides, the more decoupling is taken care of by local capacitors, the lower the board's HF emissions will be.

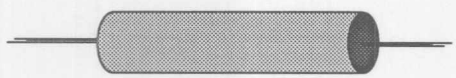
Summary

Choosing the right decoupling capacitor is an important part of high-speed circuit design. Choosing the wrong one can introduce supply noise that can slow down signal switching or even end up giving incorrect data. For decoupling Atmel PLDs, 0.22-µF capacitors are recommended. These should be of either multi-layer ceramic or plastic dielectric type. Surface-mount devices are best, with radial leaded cases also being acceptable.

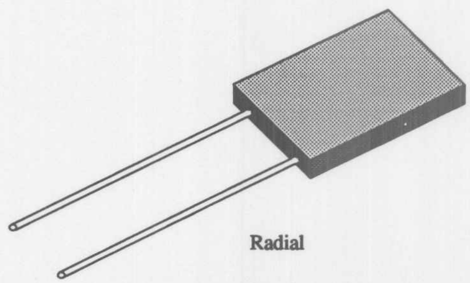
Table 1. Capacitor Types and Recommendation Ratings

Dielectric	Body	L (nH,typ)	R (ohm,typ)	Rating	Comments
Ceramic II	SMD	2	.02	E	Highly recommended
	Radial	6	.07	G	Keep leads short
	Axial	12	.07	S	Axial always = Higher L
Ceramic III	SMD	2	.04	G	C loss hot/cold/old
	Radial	6	.15+	S	
	Axial	12	.15+	X	
Plastics	SMD	2	.03	E	Hard to find
	Radial	5+	.01+	G	Get R and L data
	Axial	12+	.01+	X	
Aluminum Electrolytic	SMD	13	9.0	X	Forget it
	Radial	15+	1.5+	X	
	Axial	20	1.5	X	
Tantalum	SMD	?	3.0	X	
	Radial	10+	1.0	X	
	Axial	15+	1.0	X	

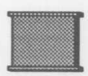
Ratings code:
E Excellent; highly recommended
G Good; will perform well in most applications
S Satisfactory; be aware of specific vendor's device performance
X Not recommended



Axial



Radial



SMD

UV Erasable Programmable Logic Device

Application Note

Using a Programmable Logic Device As a System Controller In an I/O Bus Based System [1]

Summary

As Programmable Logic Devices (PLDs) become more complex, the amount of logic that can be placed in one device is rapidly increasing. Complete controllers and subsystems now fit into one or two PLDs. As a result, the PLD may now be connected directly to the system bus as an independent peripheral. First generation PAL® devices are difficult to use in these applications. However, recent innovations in PLD architecture enable them to be easily designed into bus-based systems.

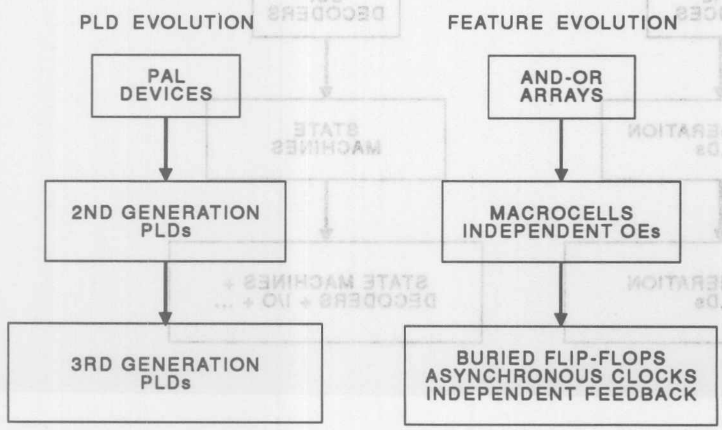
PLD Evolution

The driving force behind PLD usage has been to integrate as much of the Small Scale Integration (SSI) logic on a packed PC board as possible. The first level of integration was made possible by the invention of the PAL device. First generation products were usually in 20-pin packages with a typical device having nine dedicated inputs and eight dedicated outputs. One input pin was a dedicated output enable, and one pin a dedicated, common clock for up to eight flip-flops. Making one of these devices work on an I/O bus was difficult and typically was used as little more than a simple latch.

In the mid-eighties, second generation devices appeared. These PLDs are generally in 24 or more pins, have independent output enable controls and *output macrocells.* The macrocells allow the designer to configure each output independently as registered or combinatorial. However, there are still too few registers in these devices to allow the design of complex state machines. Also, these circuits lack independent feedback paths, which further reduce the usable number of registers. This also complicates the use of the output pins as true I/O structures.

Note: 1. This article originally appeared in Northcon '86

®PAL is the registered trademark of Monolithic Memories



5

ATV750) have appeared. These devices are differentiated by the following features:

- **Extra Registers**

Up to twice the usual number. The ATV750 has 20 flip-flops.

- **Independent Feedbacks**

Feedback paths for the registers are independent of the output configuration. In addition, there are separate input paths from the I/O pads.

- **Asynchronous Clocks**

Product terms for each flip-flop's clock allows the designer to break up the registers into different functional blocks.

Control function outputs that have no other use than to manage the other resources inside the PLD need not be brought outside the device, allowing implementation of complex state machines internally.

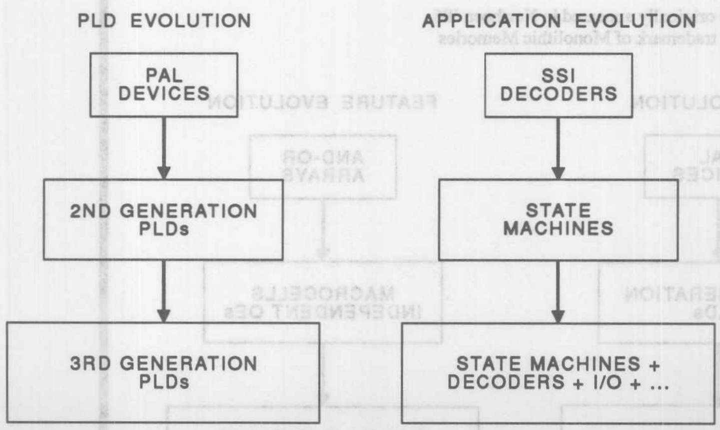
As PLDs have evolved, so have the applications for them. Initially, PLDs could only integrate a few SSI functions. A typical application was a special-purpose decoder or encoder. With the introduction of more flip-flops, Medium Scale Integration (MSI) functions such as state machines could be designed. Third generation devices are the first true Large Scale Integration (LSI) devices, and are capable of integrating several of the previous generation devices into one package. Now state machines can be combined with an output decoder to control peripheral functions, and still have adequate resources to interface directly to the microprocessor.

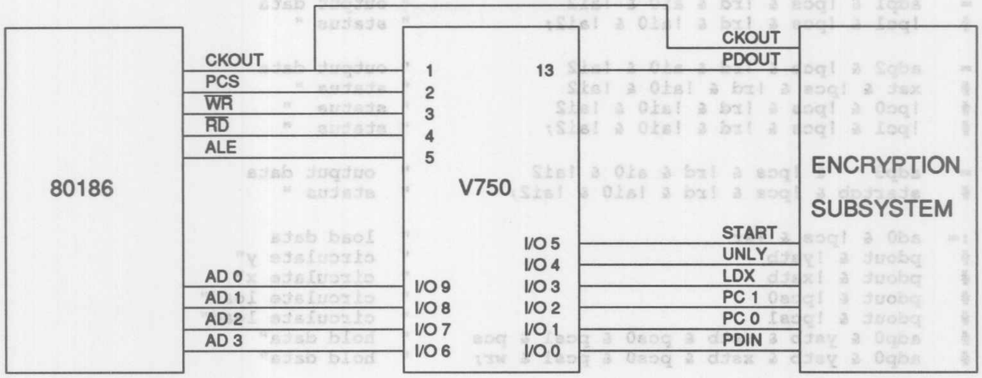
System Application

The following example is an application of the Atmel ATV750 as a peripheral resource controller. The design required a state machine, a bus interface unit and a peripheral control interface. All ten outputs of the ATV750 are used, most in the combinatorial mode. However, the 17 required flip-flops were still available to latch the address and data buses, provide a status register, and a two-bit counter. This design would require three second generation, 24-pin PLDs, or five first generation 20-pin devices and at least two other discrete devices. In all, more than 80% of the ATV750 is utilized. The number of gates alone integrated into the ATV750 in this application is more than other 24-pin PLD's have to offer.

- **The System**

The system described is a peripheral controller/bus interface for connecting a special-purpose, custom encryption / exponentiation chip to an 80186 microprocessor (Figure 1). The custom chip has a serial interface, and only one bi-directional pin to indicate its "busy" status. All chip functions are controlled with a set of single-purpose input pins. While simple, this interface is not directly compatible with a modern microprocessor, such as the 80186. The PLD system described not only combines the required glue logic, but also off-loads the parallel-to-serial conversion from the processor. This application note will only touch on the salient features of the design, and why a third generation device is so useful.





5

- **The Microprocessor Bus**

The 80186 uses a multiplexed address and data bus. Several control signals, such as ALE, $\overline{RD}$ and $\overline{WR}$ tell the system when to get what type of data from the bus. The 80186 also has some internally decoded chip selects, and one is used here for convenience. The system clock is an 8-MHz signal, which is appropriate for the encryption chip and well within this PLD's timing specification. The lower four bits of the address are latched into the PLD to define the upcoming operation, which then allows the PLD to output the requested data in one read cycle of the microprocessor. These address bits are decoded to define the instruction to be executed by the PLD subsystem.

- **Tackling the I/O Bus**

Using first and second generation PLDs, the equations for the I/O bus interface are shown in Figure 2. These equations consume twelve sum terms, eight flip-flops, and twelve output pins. Since this requires two PLDs, another ten input pins are required as well. When rewritten for the ATV750, only four macrocells are required, and eight sum terms and flip-flops. No extra inputs are required, as the ATV750's I/Os are true input/output pins.

The equations for the ATV750 are in Figure 3. This compaction is possible for three reasons:

1. The individual product terms for OE permit the pin to be used as both an input and output.

2. The three feedback paths allow both registers to be used while the pin status is still available to the array.

3. The product term for the flip-flop clock means that the sum term for one of the flip-flops can be shared between the D input and the output pin. <u>A single ATV750 macrocell can incorporate logic which would require up to three output pins and one input pin in other PLDs.</u>

Figure 2.

```
ad0   =   adp0 & !pcs & !rd & ai0 & !ai2             " output data
      #   !pc0 & !pcs & !rd & !ai0 & !ai2            " status "
      #   yst & !pcs & !rd & !ai0 & !ai2;            " status unly

ad1   =   adp1 & !pcs & !rd & ai0 & !ai2             " output data
      #   !pc1 & !pcs & !rd & !ai0 & !ai2;           " status "

ad2   =   adp2 & !pcs & !rd & ai0 & !ai2             " output data
      #   xst & !pcs & !rd & !ai0 & !ai2             " status "
      #   !pc0 & !pcs & !rd & !ai0 & !ai2            " status  "
      #   !pc1 & !pcs & !rd & !ai0 & !ai2;           " status  "

ad3   =   adp3    & !pcs & !rd & ai0 & !ai2          "  output data
      #   startqb & !pcs & !rd & !ai0 & !ai2;        "  status "

adp0  :=  ad0 & !pcs & !wr                           "  load data
      #   pdout & !ystb                              "  circulate y"
      #   pdout & !xstb                              "  circulate x"
      #   pdout & !pcs0                              "  circulate load"
      #   pdout & !pcs1                              "  circulate load"
      #   adp0 & ystb & xstb & pcs0 & pcs1 & pcs     "  hold data"
      #   adp0 & ystb & xstb & pcs0 & pcs1 & wr;     "  hold data"

adp1  :=  ad1 & !pcs & !wr                           "  load data
      #   adp0 & !ystb                               "  circulate y"
      #   adp0 & !xstb                               "  circulate x"
      #   adp0 & !pcs0                               "  circulate load"
      #   adp0 & !pcs1                               "  circulate load"
      #   adp1 & ystb & xstb & pcs0 & pcs1 & pcs     "  hold data"
      #   adp1 & ystb & xstb & pcs0 & pcs1 & wr;     "  hold data"

adp2  :=  ad2 & !pcs & !wr                           "  load data
      #   adp1 & !ystb                               "  circulate y"
      #   adp1 & !xstb                               "  circulate x"
      #   adp1 & !pcs0                               "  circulate load"
      #   adp1 & !pcs1                               "  circulate load"
      #   adp2 & ystb & xstb & pcs0 & pcs1 & pcs     "  hold data"
      #   adp2 & ystb & xstb & pcs0 & pcs1 & wr;     "  hold data"

adp3  :=  ad3 & !pcs & !wr                           "  load data
      #   adp2 & !ystb                               "  circulate y"
      #   adp2 & !xstb                               "  circulate x"
      #   adp2 & !pcs0                               "  circulate load"
      #   adp2 & !pcs1                               "  circulate load"
      #   adp3 & ystb & xstb & pcs0 & pcs1 & pcs     "  hold data"
      #   adp3 & ystb & xstb & pcs0 & pcs1 & wr;     "  hold data"

ai0   :=  ad0 & pcs                                  "  idle state "
      #   ad0 & ale                                  "  idle state "
      #   ai0 & !pcs & !ale;                         "  hold instruction"

ai1   :=  ad1 & pcs                                  "  idle state "
      #   ad1 & ale                                  "  idle state "
      #   ai1 & !pcs & !ale;                         "  hold instruction"

ai2   :=  ad2 & pcs                                  "  idle state "
      #   ad2 & ale                                  "  idle state "
      #   ai2 & !pcs & !ale;                         "  hold instruction"

ai3   :=  ad3 & pcs                                  "  idle state "
      #   ad3 & ale                                  "  idle state "
      #   ai3 & !pcs & !ale;                         "  hold instruction"
```

Figure 3.

```
ai0.ck      =    clk2 & !ale; ai2.ck  =     clk2 & !ale; "clock instruction
ai1.ck      =    clk2 & !ale; ai3.ck  =     clk2 & !ale; "clock instruction

ai0         =    ad0 & !pcs & ale                        "load instruction"
            #    adp0 & !pcs & !rd & ai0 & !ai2          "output data
            #    !pc0 & !pcs & !rd & !ai0 & !ai2         "status "
            #    yst & !pcs & !rd & !ai0 & !ai2          "status unly

ai1         =    ad1 & !pcs & ale                        "load instruction"
            #    adp1 & !pcs & !rd & ai0 & !ai2          "output data
            #    !pc1 & !pcs & !rd & !ai0 & !ai2;        "status "

ai2         =    ad2 & !pcs & ale                        "load instruction"
            #    adp2 & !pcs & !rd & ai0 & !ai2          "output data
            #    xst & !pcs & !rd & !ai0 & !ai2          "status "
            #    !pc0 & !pcs & !rd & !ai0 & !ai2         "status "
            #    !pc1 & !pcs & !rd & !ai0 & !ai2;        "status "

ai3         =    ad3 & !pcs & ale                        "load instruction"
            #    adp3   & !pcs & !rd & ai0 & !ai2        "output data
            #    startqb & !pcs & !rd & !ai0 & !ai2;     "status "

enable ad0  =    !pcs & !rd;   enable ad2  =     !pcs & !rd;
enable ad1  =    !pcs & !rd;   enable ad3  =     !pcs & !rd;
(adp equations remain the same as before, but are now buried in the macrocell)
```

Figure 4.

```
!pc0        =    !clk22 & !pcs0;

!pc1        =    !clk22 & !pcs1;

!pcs0       :=   ai2 & ai0 & start
            #    !cn0 & count & !pcs0
            #    !cn1 & count & !pcs0

 !pcs1      :=   ai1 & ai2 & start
            #    !cn0 & count & !pcs1
            #    !cn1 & count & !pcs1;
            #    !cn1 & count & !pcs1;
```

5

• The Chip Interface

The encryption chip is loaded and unloaded serially, four bits at a time in this design. The equations for the interface logic are in Figure 4. Also in this figure is a simple state diagram for the two-bit counter required for this design. This state machine is buried, and its decoded outputs are used to control the serial transfers.

• Starting the Peripheral Chip

To begin execution in the peripheral chip, a bi-directional signal named *start* is asserted. This is an active low signal. The controller must assert this signal low for four clock cycles. Then the exponentiation chip will hold this line low until it has completed its operations. An external pull up resistor is required. The internal flip-flop, whose output is named *stint*, contains the state of the peripheral. This is used to signal the microprocessor that the subsystem is busy when the processor reads the ATV750's status.

• Multiplexing Flip-Flop Inputs and I/O Pins

One I/O pin / flip-flop combination can be used to store the state of the encryption chip and to output this to the peripheral. This is accomplished by multiplexing the sum term output between the flip-flop's D input and the output buffer. The sum term and the OE product term are active to begin the encryption chip's exponentiation cycle. After the state machine counter finishes counting, the output is put into the high impedance state. If the external chip has begun its operation correctly, it will then hold the pin low. Now the state of the I/O pin is used as the D input to the flip-flop, but not output because the OE term is off. The multiplexed macrocell is in Figure 7. The following simple equations are all that is required to implement this logic:

```
enable start    =       !count;
stint           :=      !coun
# !start & count;
start.c         =       clk2;
```

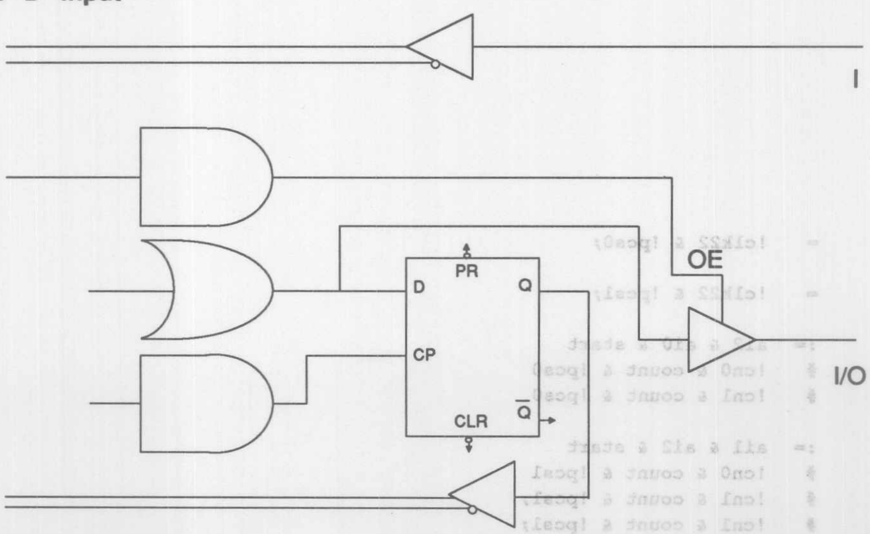
Conclusion

The application of a third generation EPLD in an I/O bus based system demonstrates the usefulness of the following features:

- **Buried Registers**
- **Independent Feedback Paths**
- **Asynchronous Register Clocks.**

This design consists of roughly 600 gates, which fit into a ATV750 gate complexity PLD with 80% utilization factor. Due to the usefulness of the new features and their implementation in the macrocell of the ATV750, this design, which would have required three second generation devices, could easily fit into one ATV750.

Mulitplexed "D" Input



UV Erasable Programmable Logic Device

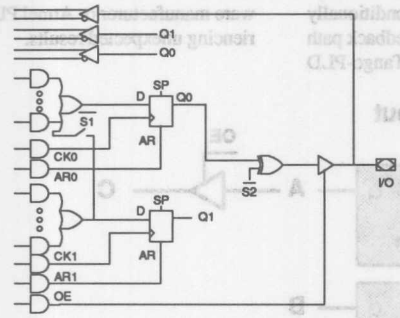
Application Note

Using the Buried Nodes and Feedbacks

Introduction

Conventional PLD I/O pin logic forces you to choose either a dedicated output pin or a dedicated input pin. This renders the output register unusable. Multiple feedback paths and individual product term controlled output enables (OEs) make the ATV750 I/O pins truly bi-directional. An ATV750 I/O pin can be configured as a dedicated input, a dedicated output, or an input and output bus interface pin. No registers are sacrificed in the process. All registers can be buried.

Registered Output



Buried Registers

To use the buried register outputs, they must have a unique name. The compilers need to know which register to associate with each signal name. Each compiler uses a different method for assigning node numbers to signal names. Table 1 shows how to assign these names for each compiler. Table 2 lists node names for each compiler. Note: Q1 need not be defined if Q0 is sharing Q1's product terms.

Abel™, Atmel-Abel™, Cupl™, and Tango-PLD: If the output is combinatorial or if Q0 shares the product terms of Q1 in a registered output, the Q1 node does not need a name. Name Q1 nodes with the proper node numbers and refer to Feedback Options on how to access Q0s and Q1s.

LOGiC: LOGiC requires you to name the Q0 nodes if you are using the pin as an input and still using Q0, or if you are using OE to make the pin an input and an output.

PistoHI: No node numbers are necessary. Q0 nodes are named by declaring

```
: nodename0 Q0! pinname;
```

Q1 nodes are named by declaring

```
: nodename1 Q1! pinname;
```

Table 1. ATV750 Node Declaration

Product	Example	Node	Declaration	Comments
Abel™	anyname	node	26;	
Atmel-Abel™	anyname	node	26;	
Cupl™	pinnode	25 =	anyname;	
LOGiC	anyname	=	1;	Following key word *NODE
PistoHI	anyname	Q0!	pinname;	Use Q1! for Q1 declaration
TangoPLD	anyname			In PUTPART place "anyname" as the 25th signal

Table 2. ATV750/ATV750B Node Numbers

ATV750 Pin Numbers	Abel™ Q1	Atmel-Abel™ Q1	Cupl™ Q1	LOGiC Q0	LOGiC Q1	TangoPLD Q1
14	26	26	25	2	1	25
15	27	27	26	4	3	26
16	28	28	27	6	5	27
17	29	29	28	8	7	28
18	30	30	29	10	9	29
19	31	31	30	12	11	30
20	32	32	31	14	13	31
21	33	33	32	16	15	32
22	34	34	33	18	17	33
23	35	35	34	20	19	34

ATV750 Feedbacks

Each third party product uses a slightly different syntax for accessing the feedback paths (refer to Table 3). Whenever the Q0 register is used and the OE term is disabled or conditionally disabled, care must be taken to ensure the correct feedback path is referred to in your equations. Version 1.11 of Tango-PLD does not support the Q0 feedback. This syntax may change with each new software revision. Please check with the specific software manufacturer or Atmel PLD Applications if you are experiencing unexpected results.

Feedback Options: Registered Output

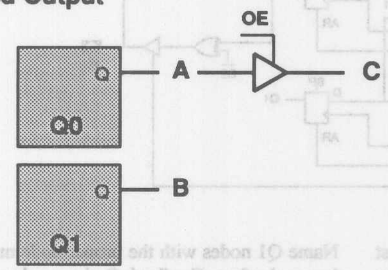


Table 3. ATV750/ATV750B Feedback Paths

	A	B	C
Abel™	pinname.FB	nodename	pinname
Atmel-Abel™	pinname.FB	nodename	pinname
Cupl™	pinname	nodename	pinname.IO
LOGiC	nodename0	nodename1	pinname
PistoHl	nodename0	nodename1	pinname
TangoPLD	no support	nodename	pinname

ATV2500 Node Numbering

With an additional OR, the ATV2500 logic cell becomes even more versatile than the ATV750 logic cell. Under certain situations, an additional set of buried registers must be defined. The same syntax used for ATV750 (Table 3) can be used to name the ATV2500 buried registers. The node numbers are listed in Table 4.

Abel™, Atmel-Abel™, Cupl™, and TangoPLD: Q1 need not be named when the output logic cell is configured as 8- or 12-term combinatorial output. Q2 need not be named when output logic cell is configured as 12-product term combinatorial or registered output. Name Q1 and Q2 nodes with the proper node numbers and refer to Feedback Options for selecting the correct feedback paths.

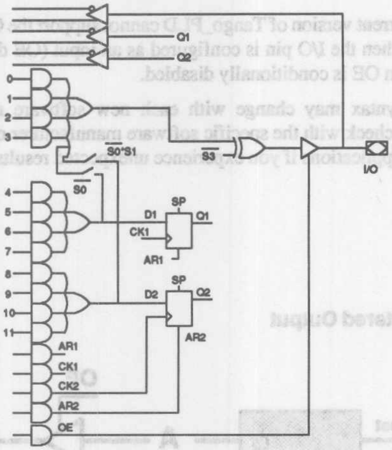
PistoHl: No node numbers are necessary. Q1 nodes are named by declaring

```
: nodename1 Q1! pinname;
```

Q2 nodes are named by declaring

```
: nodename2 Q2! pinname;
```

Combinatorial Output



Registered Output

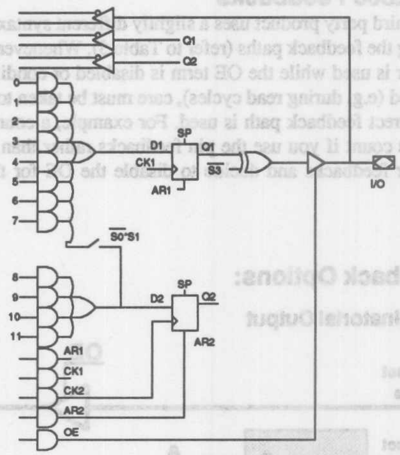


Table 4. ATV2500/ATV2500B Node Numbers

ATV2500 Pin Numbers	Abel™		Atmel-Abel™		Cupl™		TangoPLD	
	Q1	Q2	Q1	Q2	Q1	Q2	Q1	Q2
4[1]	217	41	217	41	65	41	41	65
5[1]	218	42	218	42	66	42	42	66
6[2,3]	219	43	219	43	67	43	43	67
7[3]	220	44	220	44	68	44	44	68
8	221	45	221	45	69	45	45	69
9	222	46	222	46	70	46	46	70
11	223	47	223	47	71	47	47	71
12	224	48	224	48	72	48	48	72
13	225	49	225	49	73	49	49	73
14	226	50	226	50	74	50	50	74
15	227	51	227	51	75	51	51	75
16	228	52	228	52	76	52	52	76
24	229	53	229	53	77	53	53	77
25	230	54	230	54	78	54	54	78
26	231	55	231	55	79	55	55	79
27	232	56	232	56	80	56	56	80
28	233	57	233	57	81	57	57	81
29	234	58	234	58	82	58	58	82
31	235	59	235	59	83	59	59	83
32	236	60	236	60	84	60	60	84
33	237	61	237	61	85	61	61	85
34	238	62	238	62	86	62	62	86
35	239	63	239	63	87	63	63	87
36	240	64	240	64	88	64	64	88

Notes:
1. Due to the memory limitations of PC/MS DOS, Abel™ PC versions 3.0 to 3.2 and Atmel-Abel™ version 1.01 do not support the macrocells associated with pin 4 and 5. These pins can only be used as inputs.
2. These same versions of Abel™ and Atmel-Abel™ (see above) do not support the AR terms of Q2 associated with pin 6.
3. These same versions of Abel™ and Atmel-Abel™ (see above) do not support the Synchronous Preset of pin 6 and 7.

ATV2500 Feedbacks

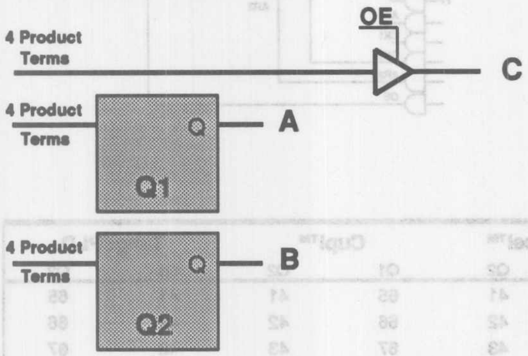
Each third party product uses a slightly different syntax for accessing the feedback paths (refer to Table 5). Whenever the Q1 register is used while the OE term is disabled or conditionally disabled (e.g. during read cycles), care must be taken to ensure the correct feedback path is used. For example, a counter can lose its count if you use the pin feedbacks rather than the Q1 register feedbacks and decide to disable the OE for the read cycle.

The current version of Tango_PLD cannot support the Q1 feedback when the I/O pin is configured as an input (OE disabled) or when OE is conditionally disabled.

This syntax may change with each new software revision. Please check with the specific software manufacturer or Atmel PLD Applications if you experience unexpected results.

Feedback Options:

Combinatorial Output



Registered Output

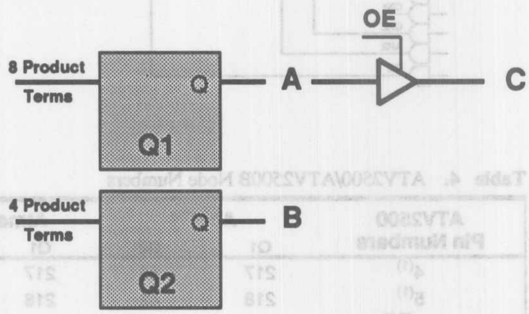


Table 5. ATV2500/ATV2500B Feedback Paths

		A	B	C
Abel™	Registered	pinname.FB	nodename2	pinname
	Combinatorial	nodename1	nodename2	pinname,
Cupl™	Registered	pinname	nodename2	pinname.IO
	Combinatorial	nodename1	nodename2	pinname
Tango-PLD	Registered	no support	nodename2	pinname
	Combinatorial	nodename1	nodename2	pinname

Testing Non-Windowed OTP PLDs

Atmel's testing of non-windowed OTP PLDs is comprehensive and thorough. It is sufficient to guarantee programmability and performance. The wafer-probe test checks 100% of all memory elements for programmability. Final test of packaged units checks programming a second time. Performance testing on the *Quality Test Array* (QTA) guarantees AC test parameters. This data shows a high degree of correlation with standard array performance data.

Introduction

Atmel's corporate goal is to meet or exceed our customers' requirements 100% of the time. This means we must prove to ourselves that each product shipped will perform as specified or better.

PLDs must meet two different device requirements: they must program as a memory device, and they must function as a logic device. This requires testing all devices in two very different ways.

The programmable elements in Atmel's OTP PLDs are UV EPROM memory cells. The AND array programs like an EPROM. UV light cannot penetrate non-windowed packages (e.g., any plastic package or solid lid ceramic package). Erasing a non-windowed EPROM element is not possible. Any testing by Atmel before shipment cannot program the main AND array or the customer cannot enter his own pattern.

Atmel's test methodology guarantees our PLDs will program and perform to the data sheet, even without programming this main AND array.

Figure 1. Non-Windowed OTP Test Flow

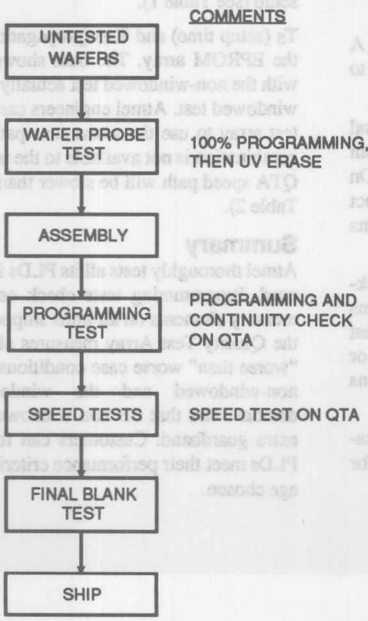
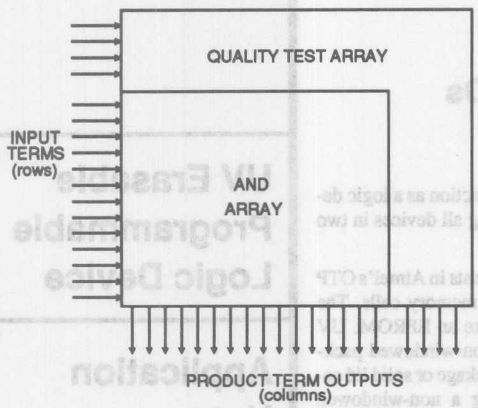


Figure 2. PLD Topography



Programming

After wafer fabrication, the first test is wafer-probe (Figure 1). All PLD dice manufactured by Atmel use a similar wafer test program. This test programs one-hundred percent of the memory bits used to implement the AND array in our PLDs. This programming algorithm is more aggressive than that implemented by programming manufacturers. After verifying all bits, the wafer is then UV erased before proceeding further.

Package assembly is next. If the packaging used does not have a UV translucent window, the customer is the only one who can program the device. Atmel must guarantee performance and programming by testing other EPROM cells in the device.

These other EPROM cells are next to the main AND array. A discussion of the PLD AND array topography clarifies how to test the extra EPROM cells (Figure 2).

Product term inputs enter the AND array from one end, and form the *rows* of the matrix. Product term outputs leave from the bottom— product terms form the *columns* of the matrix. On the far right of Figure 2 are the Quality Test Array product terms (columns). At the top of the array are the QTA input terms (rows).

Even though wafer-probe tests all units' programming, packaged units also have their QTA terms tested. Since these terms are physically at the ends of the inputs and outputs, this test checks the integrity of these lines. The test detects any shorts or opens on the inputs or outputs. Programming these locations also provides a thorough test of the programming circuitry.

The wafer-probe test programs one-hundred percent of all locations, and the package test checks the programming circuitry for continuity and programmability.

Performance Testing

The logic test parameters divide into two groups: DC and AC test values. All of the DC test parameter tests are independent of the logic programmed into the device. This means that DC testing is the same for every PLD sold by Atmel, independent of the package used.

Every different logic implementation results in different AC performance. Atmel measures AC performance using a "worst case" pattern. This pattern is slightly different in nature for each PLD. This pattern uses every resource available in the device. Outputs can be made to switch in the same direction at the same time. The test program applies worst case combinations of input term and product term patterns to the device. Atmel collects AC data with this pattern on each UV erasable device shipped.

Non-erasable devices cannot use their main array for this test. Atmel programs a special speed test pattern into the QTA. Tests on this array measure all of the same AC characteristics as those done on erasable units. Correlations between the main array and the QTA follow.

Correlation

Three parameters measured on 26 units show a high degree of correlation between windowed PLD testing and non-windowed PLD testing. Measurements of T_{CO}, T_S, and T_{PD} show little difference between performance testing on the standard array versus performance testing on the QTA.

T_{CO} (clock to output) is independent of the memory array. The signal path is directly from the clock input buffer to all ten flip-flops in the AT22V10, and then straight to the output buffer. The data shows a very tight distribution, and the difference in the two measurement techniques is within the accuracy of the setup (see Table 1).

T_S (setup time) and T_{PD} (propagation delay) are dependent on the EPROM array. The data shows an excellent correlation, with the non-windowed test actually measuring slower than the windowed test. Atmel engineers carefully designed the quality test array to use the worst case paths for all conditions. This combination is not available to the user, but guarantees that the QTA speed path will be slower than the user's speed path (see Table 2).

Summary

Atmel thoroughly tests all its PLDs independent of the package used. Programming tests check one-hundred percent of the memory elements on all units shipped. Performance testing on the Quality Test Array measures all AC test parameters with "worse than" worse case conditions. Data taken with both the non-windowed and the windowed performance tests demonstrates that the non-windowed test is accurate and has extra guardband. Customers can feel confident that Atmel's PLDs meet their performance criteria independent of the package chosen.

Table 1. T_{CO} Performance Testing Comparison

Parameter Test	T_{CO} Window	T_{CO} Non-Window	Delta
Average	8.91 ns	8.68 ns	+0.23 ns
Std. Dev.	0.18 ns	0.15 ns	0.06 ns

Table 2. T_S and T_{PD} Performance Testing Comparison

Parameter Test	T_S Window	T_S Non-Window	Delta	T_{PD} Window	T_{PD} Non-Window	Delta
Average	6.80 ns	6.97 ns	-0.17 ns	13.92 ns	14.17 ns	-0.25 ns
Std. Dev.	0.09 ns	0.11 ns	0.09 ns	0.25 ns	0.25 ns	0.18 ns

Designing with the Atmel-ViewPLD Development Tool

UV Erasable Programmable Logic Device

Application Note

Like the Atmel-Abel™ software, the Atmel-ViewPLD development tool uses a popular industry-standard CAE development system. The development tool integrates the Viewlogic Workview software as the design environment with Data I/O's back-end technology of device fitting and fusemapping. With this development tool, a designer can use schematic capture and/or the Abel™ Hardware Description Language (Abel HDL) to functionally describe his or her design requirements. Additionally, the development tool also allows the designer's previous PLD designs in the form of JEDEC files to be read and modelled so that the schematic representations of the designs can be generated.

In this application note, the Atmel-ViewPLD features developed specifically for Atmel PLDs are discussed. Also included are the design hints which allow your designs to be efficiently fitted into your favorite Atmel PLD, and the Schematic-to-JEDEC and Abel-to-JEDEC compilation process flows. Below is a listing of the topics covered in this application note:

- The Atmel-ViewPLD Menu Commands "Schematic → JEDEC," "Abel → JEDEC," and "Prep WIR (Device)"
- The DIO and DSTD symbols
- Using the IN.1 and OUT.1 symbols for your design input and output pins
- Setting your input signal to a constant logic level
- Applying the D_LATCH.1 and IN_LATCH.1 symbols for the ATV5000 or ATV5100 devices
- What to do with the unused pins of a DSTD symbol
- Creating a Viewdraw schematic from an Abel™ or a JEDEC file
- Simulating your design using worst-case timing values

For your reference, the DIO primitive and DSTD symbols are listed at the end of this application note.

The Atmel-ViewPLD Menu Commands

"Schematic → JEDEC," "Abel → JEDEC," and "Prep WIR (Device)"

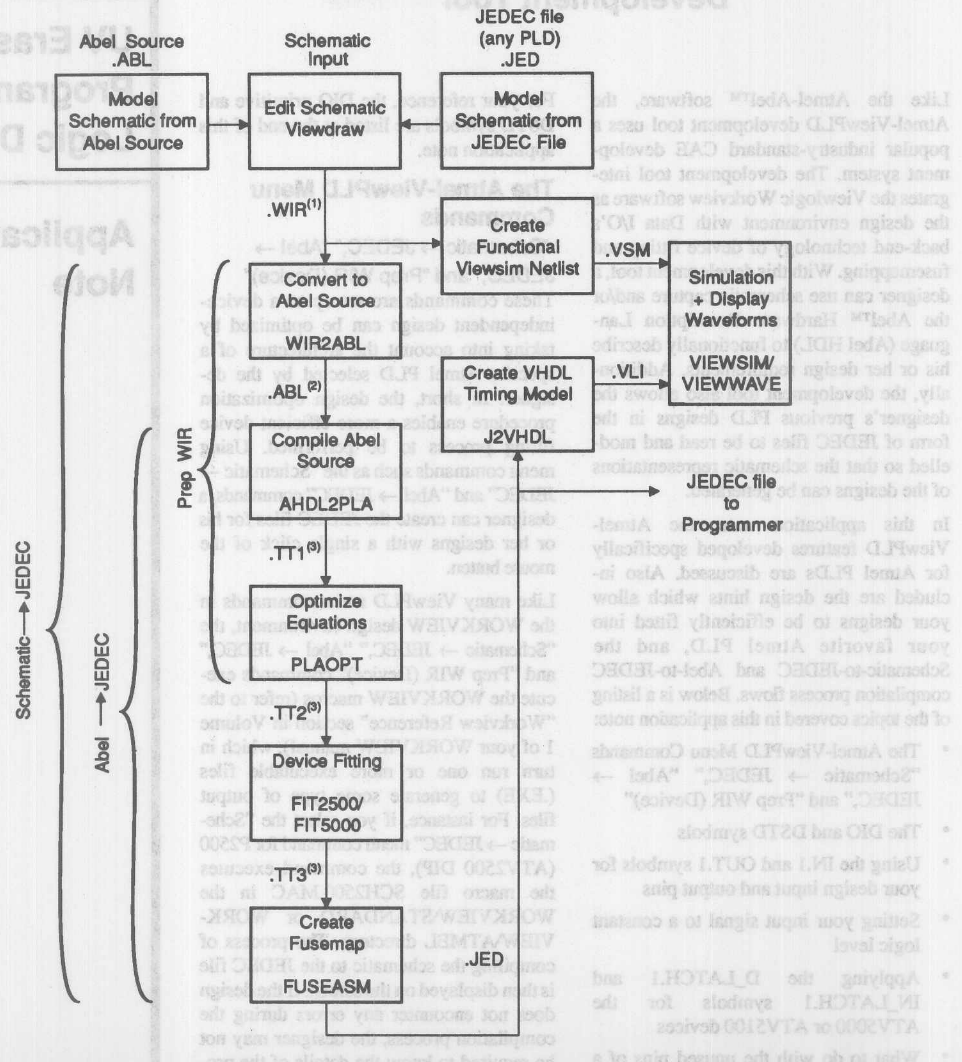
These commands are set up so a device-independent design can be optimized by taking into account the architecture of a specific Atmel PLD selected by the designer. In short, the design optimization procedure enables a more efficient device fitting process to be performed. Using menu commands such as the "Schematic → JEDEC" and "Abel → JEDEC" commands, a designer can create the JEDEC files for his or her designs with a single click of the mouse button.

Like many ViewPLD menu commands in the WORKVIEW design environment, the "Schematic → JEDEC," "Abel → JEDEC," and "Prep WIR (Device)" commands execute the WORKVIEW macros (refer to the "Workview Reference" section in Volume 1 of your WORKVIEW manual), which in turn run one or more executable files (.EXE) to generate some type of output files. For instance, if you select the "Schematic → JEDEC" menu command for P2500 (ATV2500 DIP), the command executes the macro file SCH2500.MAC in the WORKVIEW\STANDARD or WORKVIEW\ATMEL directory. The process of compiling the schematic to the JEDEC file is then displayed on the screen. If the design does not encounter any errors during the compilation process, the designer may not be required to know the details of the process. However, it would be advantageous for the designer to understand each process step, especially when the design encounters some compilation errors.

5

open your design schematic via Viewdraw. These commands be prompted for immediately after execution of the command.

Figure 1. The Atmel-ViewPLD Compilation Process



Notes:

1. A WIR file is created by Viewdraw only after executing a "File Write" command in Viewdraw.
2. Before generating the Abel™ (.ABL) file, the WIR2ABL (WIR-to-Abel) generates an intermediate file with extension .VNT. If an error occurs in the WIR-to-Abel conversion specifying that there is an intermediate file error, then the error description will listed in this .VNT file.
3. The files .TT1, .TT2, and .TT3 are in the Open Abel PLA format.

The DIO and DSTD Symbols

Like all other CAE primitive symbols, the DIO primitive symbols provide the building blocks for implementing an Atmel-ViewPLD design. In the Atmel-ViewPLD development tool, all of the functional DIO symbols such as AND2, AND4, DFF, and TFFC, are logically described using Boolean equations in the functional library files. These files, which include V2500.FLB, V5000.FLB, and VIEWLOGI.FLB, are located in the WORKVIEW\STANDARD\LIB4 or WORKVIEW\ATMEL\LIB4 directory. Figure 2 illustrates the AND2 and TFF symbols and their logic representations in the functional library file.

As shown in Figure 2, the architectural capabilities in each Atmel PLD determine the type of equation in the function library file. In fact, there are some DIO symbols which are not applicable for some of the Atmel PLDs because of their architecture limitations. For instance, the DFFPC symbol, which is a D-type flip-flop with asynchronous preset and clear controls, is applicable for the ATV5000 and ATV5100 devices, but not for the ATV750 and ATV2500 devices. Please refer to the DIO Library listing (Table 2) at the end of this application note for the complete information on the applicabilities of the DIO symbols.

Table 1 shows the three functional library files included with the Atmel-ViewPLD development tool:

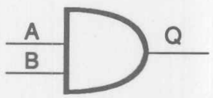
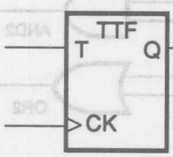
Table 1. Function Library Files

FLB File	Description
V2500.FLB	Logic equations optimized for ATV750 and ATV2500 devices
V5000.FLB	Logic equations optimized for ATV5000 and ATV5100 devices
Viewlogi.FLB	Generic library called by the "PREP WIR" command, not the "Prep WIR (Device)" command

The first two library files are optimized for the ATV750, ATV2500, and ATV5000/5100 device families. The third file, VIEWLOGI.FLB, is used by the "Prep WIR" menu command. This command is designed for use if the designer wishes to partition his or her design into multiple designs which are optimized for the devices selected by the designer. Since each functional library file is in ASCII format, the logic equations representing the DIO symbols can easily be modified by the designer. This flexibility allows the designer to generate his or her own DIO primitive symbols.

For more information on the DIO and DSTD symbols, please refer to the listings found in Table 3 at the end of this application note. Note that the DSTD symbols are the 74-Series TTL symbols which are made up of the DIO primitive symbols.

Figure 2. The AND2 and TFF DIO Symbols and their Logic Equations in the V2500.FLB and V5000.FLB Files

DIO Symbol	Logic Equation in V2500.FLB	Logic Equation in V5000.FLB
	`Q = A & B;`	`Q = A & B;`
	`Q = (T & !Q)# (!T & Q); Q.ck = CK;`	`Q = T; Q.ck = CK;`

Using the IN.1 and OUT.1 Symbols for your Design Input and Output Pins

For creating the input and output pins for your design, you would need to add the IN.1 and OUT.1 symbols (see Figure 3) from the DIO library to your design. Simply use the "Add Comp" menu command to add the IN.1 or OUT.1 symbol to the dangling input nodes. Pin labels must also be assigned to these symbols, and not assigned to the nets or nodes. For a design to be compiled successfully, you must have at least one input pin defined. Note that a design will simulate properly via Viewsim without any IN.1 and OUT.1 symbols (see Figures 3-4).

Figure 3. IN.1 and OUT.1 DIO Symbols[1]

Symbols

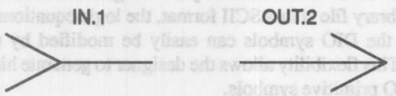
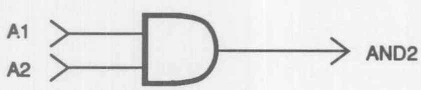


Figure 4. Design Example using IN.1 and OUT.1 Symbols

Schematic



Abel™ Source File

```
AND2, A2, A1 pin;
AND2 istype 'com';
```

Equations

```
AND2 = A1 & A2;
```

Notes:

1. Please refer to the next section when you are asssigning the VDD and GND labels to the IN.1 and OUT.1 symbols.

Setting your Input Signal to a Constant Logic Level

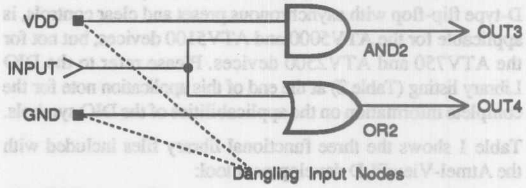
There are two methods which you can use to set your input signals to a constant logic level:

Method 1. Label your input nodes using the VDD and GND labels (see Figure 5).

Method 2. Use the PWR.1 and GND.1 symbols from the DIO library (see Figure 6).

Figure 5. Design Example using VDD and GND Labels[2]

Schematic



Abel™ Source File

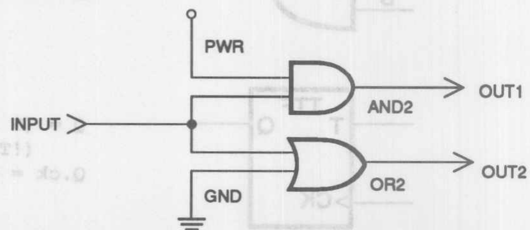
```
OUT3,OUT4,INPUT pin;
OUT3,OUT4 istype 'com';
```

Equations

```
OUT3 = 1 & INPUT;
OUT4 = 0 # INPUT;
```

Figure 6. Design Example using PWR.1 and GND.1 DIO Symbols

Schematic



Abel™ Source File

```
OUT2,OUT1,INPUT pin;
OUT2,OUT1 istype 'com';
```

Equations

```
OUT2 = INPUT # 0;
OUT1 = 1 & INPUT;
```

2. Do not use the IN.1 and OUT.1 symbols when using the VDD and GND labels to create a constant logic level. If you use these input and output symbols, then the external pins with labels "VDD" and "GND" will be generated.

Applying the D_LATCH.1 and IN_LATCH.1 Symbols for the ATV5000 or ATV5100 Device

The D_LATCH.1 symbol can be used in a design for latching internal or external signals. This symbol uses a D-type flip-flop with asynchronous preset and reset controls to emulate the latch function (the logic equations are specified in the functional library files; refer to "The DIO and DSTD Symbols" section for more information). This symbol can be used as an output or buried latch. The data flows through when $\overline{G}$ is HIGH. Once $\overline{G}$ goes LOW, the data is latched (see Figure 7).

The IN_LATCH.1 symbol is only used for latching inputs in the ATV5000 or ATV5100 design. Unlike the D_LATCH.1 symbol, this symbol is implemented in the ATV5000 or ATV5100 using a real latch circuitry in each I/O pin. The data flows through when C is LOW, and data is captured on the rising edge of the C signal (see Figure 8).

What to Do with the Unused Pins of a DSTD Symbol[1]

For DSTD symbols consisting of combinatorial logic such as the DSTD155 or DSTD156, the unused pins can be left unconnected. The unused logic will be automatically removed from the Abel™ source file as illustrated by Design Example 1 in Figure 9.

On the other hand, careful attention must be paid to the unused pins if the symbols have registered or latched logic. With symbols that have registered or latched logic, pins and nodes are automatically assigned to the unused inputs and outputs respectively. With the limited number of pins and nodes in the most devices, the assigned pins and nodes may prevent your final design from fitting into the selected device. Design Example 2 in Figure 10 illustrates the problem in which three pins and nodes are unnecessarily created.

In Design Example 2 (Figure 10), a total of three unnecessary pins were assigned and three unnecessary nodes generated.

To eliminate all the unnecessary assigned pins, we recommend grounding the unused inputs. To reduce the unnecessary assigned nodes to a single node, all the unused outputs of the symbol should be connected together to form a single node with an easily identifiable label such as NC or DUMMY. With these modifications implemented, instead of three pins and three nodes wasted as illustrated in Design Example 2, the efficiency of your design improved with:

- no additional pins generated, and
- only one additional node created. Usually this single node would not inhibit the design from fitting into the selected device. But if it does, then it can be manually deleted off the Abel™ source file.

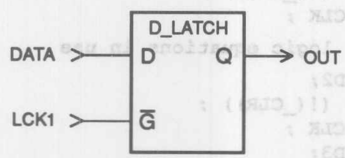
In Design Example 3 (Figure 11), only one node (i.e., node NC1) was created and no additional input pins assigned.

Note:
1. This restriction for the unused pins of the DSTD symbols will be eliminated in the future versions of the Atmel-ViewPLD development tool.

5

Figure 7. Design Example using D_LATCH.1 DIO Symbol

Schematic

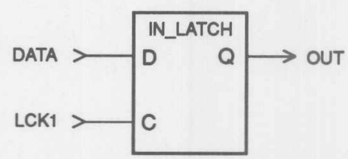


Abel™ Source File

```
OUT1,LCK1,DATA pin;
OUT istype 'buffer,reg_d';
Equations
OUT.D = 1
OUT.AP = (LCK1 & DATA);
OUT.AR = (LCK1 & !DATA);
OUT.C = 1;
```

Figure 8. Design Example using IN_LATCH.1 DIO Symbol

Schematic

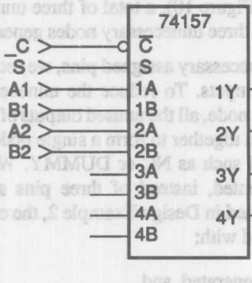


Abel™ Source File

```
OUT,LCK1,DATA pin;
OUT istype 'buffer,reg_d';
Equations
OUT.D = DATA;
OUT.LE = LCK1;
```

Figure 9. Design Example 1, A DSTD Symbol with Combinatorial Logic in which the unused pins are left unconnected.

Schematic

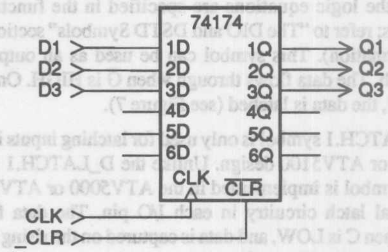


Abel™ Source File

```
 C,Y2,Y1,S,B2,B1,A2,A1 pin ;
Y1   istype   'com';
Y2   istype   'com';
Equations
Y1 = !((!(A1 & !S) & !(B1 & S)) # _C);
Y2 = !((!(A2 & !S) & !(B2 & S)) # _C);
```

Figure 10. Design Example 2, A DSTD Symbol with Registered Logic in which the unused pins are left unconnected generated.

Schematic



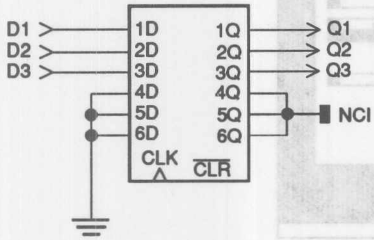
Abel™ Source File

```
S10,S7,S4 pin ; "Unused inputs
S3,S6,S9 node ; "Unused outputs
_CLR, Q3, Q2, Q1, D3, D2, D1, CLK pin ;
S9,S6,S3 istype 'buffer, reg, reg_d';
Q3,Q2,Q1 istype 'buffer, reg, reg_d';

Equations
"Unnecessary logic equations
S9.d = S10;
S9.ar = (!(_CLR)) ;
S9.c = CLK ;
S6.d = S7;
S6.ar= (!(_CLR)) ;
S6.c = CLK ;
S3.d = S4;
S3.ar = (!(_CLR)) ;
S3.c = CLK ;
"Actual logic equations in use
Q2.d = D2;
Q2.ar = (!(_CLR)) ;
Q2.c = CLK ;
Q3.d = D3;
Q3.ar = (!(_CLR)) ;
Q3.c = CLK ;
Q1.d = D1;
Q1.ar = (!(_CLR)) ;
Q1.c = CLK ;
```

tered Logic (same as in Design Example 2), but with the unused inputs grounded and unused outputs connected together to form a single node.

Schematic



Abel™ Source File

```
NC1 node ; "Unused node
_CLR,Q3,Q2,Q1,D3,D2,D1,CLK pin ;
NC1   istype   'buffer, reg, reg_d';
Q2   istype   'buffer, reg, reg_d';
Q3   istype   'buffer, reg, reg_d';
Q1   istype   'buffer, reg, reg_d';
Equations
"Unused logic equations
NC1.d = 0;
NC1.ar = (!(_CLR)) ;
NC1.c = CLK ;
"Actual logic equations in use
Q2.d = D2;
Q2.ar = (!(_CLR)) ;
Q2.c = CLK ;
Q3.d = D3;
Q3.ar = (!(_CLR)) ;
Q3.c = CLK ;
Q1.d = D1;
Q1.ar = (!(_CLR)) ;
Q1.c = CLK ;
```

Abel™ or a JEDEC File

One of the valuable features of the Atmel-ViewPLD development tool is the ability to generate a Viewdraw schematic from an Abel™ or a JEDEC file. With this feature, you can use both the schematic capture and Abel HDL design entries in the same design. In addition, old designs which consist of PLDs or TTL devices that are already in production can easily be integrated into a single design so that it can be compiled to fit into a single PLD like the Atmel ATV5000 or ATV5100 device.

The command for generating a Viewdraw schematic from an Abel™ or a JEDEC file is the "Window Open ViewPLD Model_Device Schematic" menu command. Once this command is executed, the dialog box shown in Figure 12 will open. After selecting your options, click the middle button of the mouse to begin executing the command.

Use the left button of the mouse to select the options or to enter the project name which is the Abel™ or JEDEC filename. Note that you must not enter the file extension (.ABL or JED) when entering the project name.

Figure 13 shows the process flow for generating a schematic from an Abel™ file or a JEDEC file.

5

Figure 12. Dialog Box for Modelling an Abel™, JEDEC, or PLA File to Create a Viewdraw Schematic

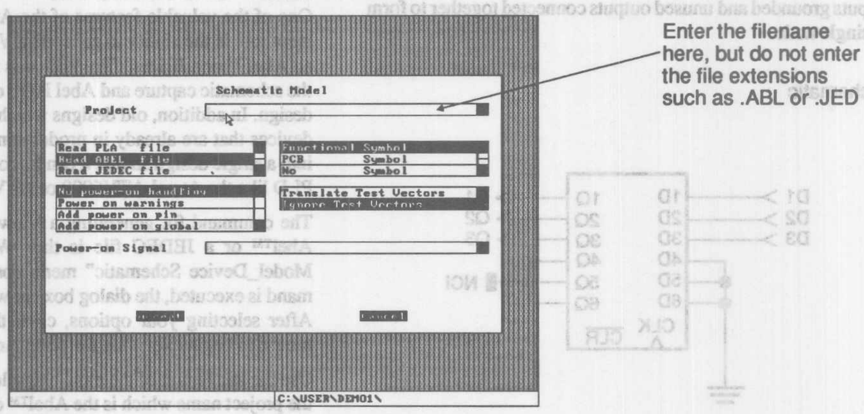
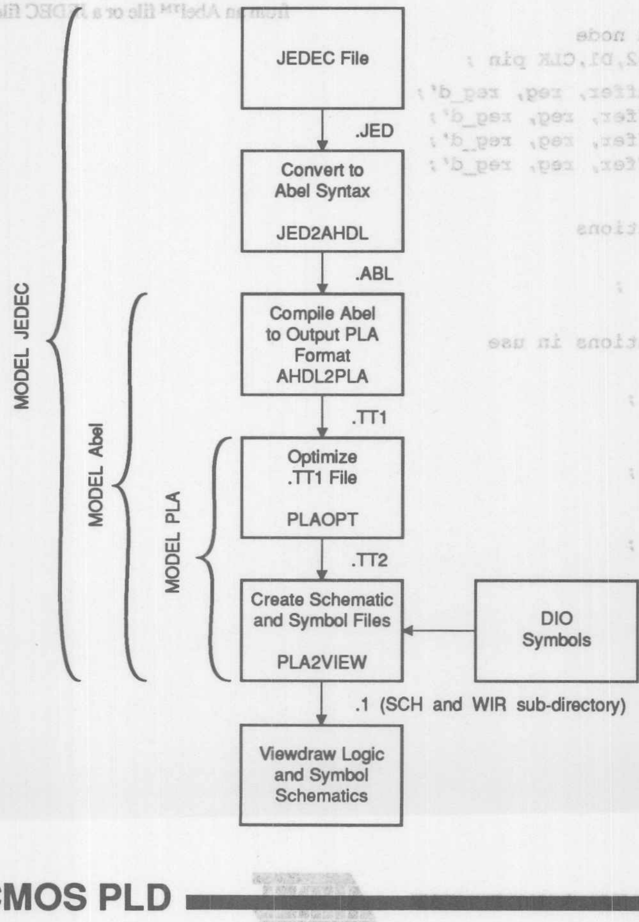


Figure 13. Process Flow for Modelling a Viewdraw Schematic



Simulating Your Design Using Worst-Case Timing Values

After you have created a JEDEC file for your design, you can further verify your design by simulating your design using the worst-case timing values of the selected device. Simulating your design using worst-case timing values provides significant benefits such as predicting the performance of your design, providing information on timing violations, and etc.

Prior to creating a timing model, you must first run the menu command "Window Open ViewPLD Search_TimBase." This command will extract the timing data of the selected device from the global timing database and store them locally in the .TIM file (same filename as your design file but with extension .TIM). Note that the ASCII timing data in the .TIM file can be modified by the users if the new timing data for a new speed version device is available from Atmel.

After creating the local timing database, execute the "Window Open ViewPLD Model_Device Timing_Model" menu command to open the JEDEC Modeller dialog box as shown in Figure 14. Select your options and click the middle button of the mouse to start executing the command (see Figure 15).

Figure 14. JEDEC Modeller Dialog Box for Creating a Timing Model with Worst-Case Timing Values

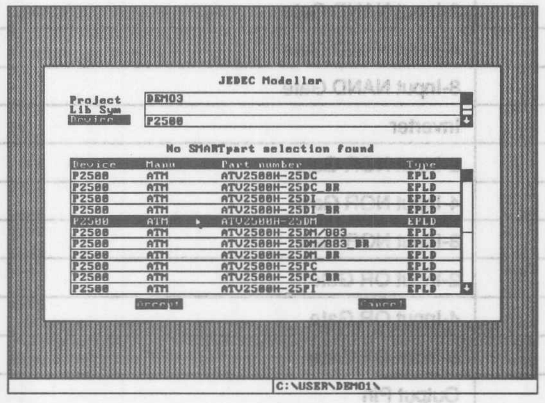
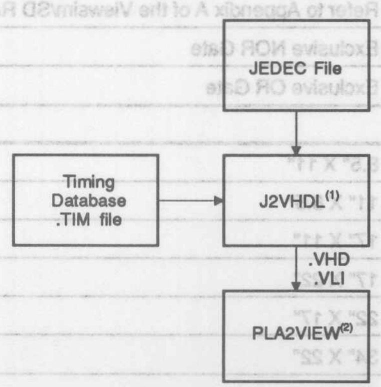


Figure 15. Process Flow for Creating a Timing Model



Notes:

1. Converts the JEDEC file to a timing model in VHDL format. The VHDL Analyzer is then executed to convert the VHDL model to a VLI model which can be read by Viewsim.
2. Creates a symbol schematic of the modelled device, i.e., only used pins will be generated.

5

Table 2. Atmel-ViewPLD DIO Primitive Symbols

Primitive	Device Supported	Description
Basic Gates		
AND2	All	2-Input AND Gate
AND4	All	4-Input AND Gate
AND8	All	8-Input AND Gate
BUF	All	Buffer
GND	All	Ground
IN	All	Input Pin
NAND2	All	2-Input NAND Gate
NAND4	All	4-Input NAND Gate
NAND8	All	8-Input NAND Gate
NOT	All	Inverter
NOR2	All	2-Input NOR Gate
NOR4	All	4-Input NOR Gate
NOR8	All	8-Input NOR Gate
OR2	All	2-Input OR Gate
OR4	All	4-Input OR Gate
OR8	All	8-Input OR Gate
OUT	All	Output Pin
PWR	All	Power
TRST	All	Tri-State Buffer, Output Enabled on High
TRIBUF	—	Viewlogic Built-in Primitive – DO NOT USE IN SCHEMATIC. Refer to Appendix A of the Viewsim/SD Reference Guide.
XNOR2	All	Exclusive NOR Gate
XOR2	All	Exclusive OR Gate
Block (Sheet) Size		
ASHEET.1	All	8.5" X 11"
ASHEET.2	All	11" X 8.5"
BSHEET	All	17" X 11"
CSHEET.1	All	17" X 22"
CSHEET.2	All	22" X 17"
DSHEET	All	34" X 22"
ESHEET.1	All	34" X 44"
ESHEET.2	All	44" X 34"

Table 2. Atmel-ViewPLD DIO Primitive Symbols (continued)

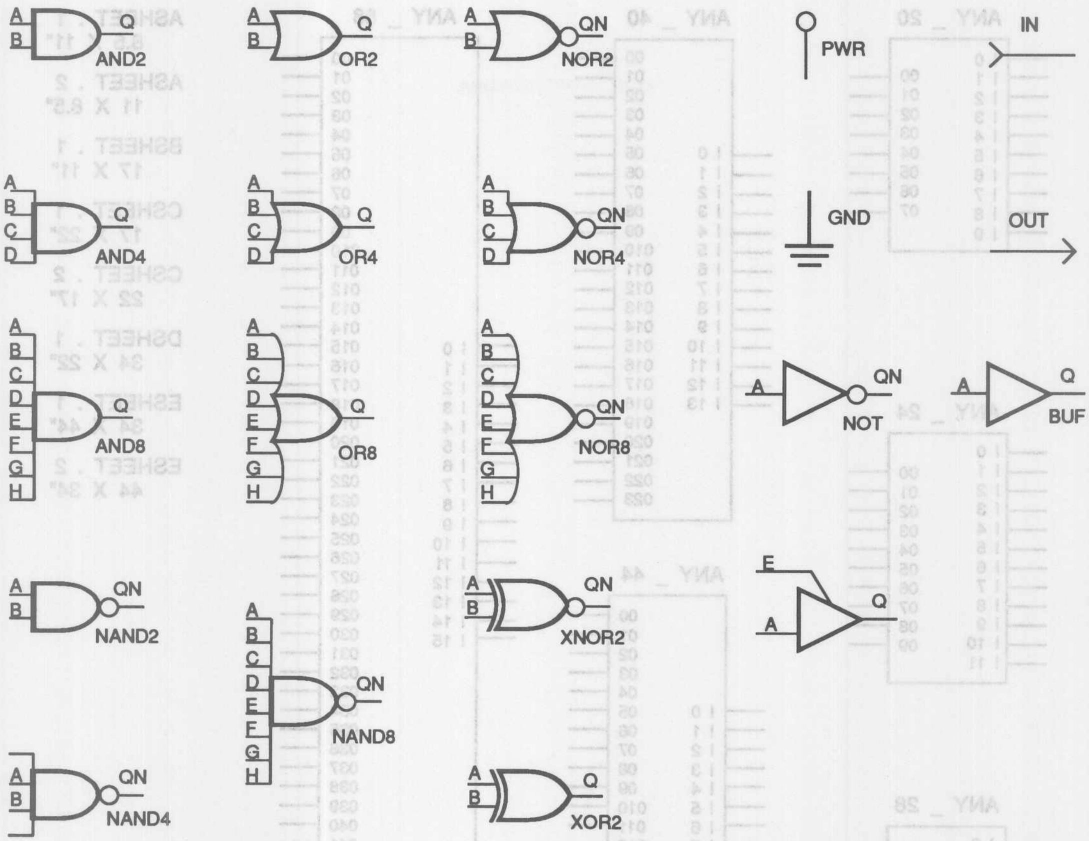
Primitive	Device Supported	Description
Generic Device Symbol (For Design Partitioning)		
ANY_20		20-Pin Generic Device
ANY_24		24-Pin Generic Device
ANY_28		28-Pin Generic Device
ANY_40		40-Pin Generic Device
ANY_44		44-Pin Generic Device
ANY_68		68-Pin Generic Device
Latches		
DLATCH	ATV5000 ATV5100	Standard D Latch with Data Flow-Through when G Input is High. Data is latched when G goes LOW. This latch can be used as a buried latch.
IN_LATCH	ATV5000 ATV5100	Input D Latch. The data flow-through occurs when the C input is LOW. Data is latched on the rising edge of C.
RSLATCH	All	RS Latch. In ATV750 and ATV2500 devices, the RS latch is emulated via two cross-coupled NAND gates. In the ATV5000 devices, a flip-flop with asynchronous preset and reset is used.
RS_FF	All	RS Flip-Flop. A D flip-flop is used to emulate the RS flip-flop.
Flip-flops		
DFF	All	D Flip-Flop with no Preset or Reset Function. Flip-flop is reset upon device power-up.
DFFC	All	D Flip-Flop with Asynchronous Reset
DFFP	ATV5000 ATV5100	D Flip-Flop with Asynchronous Preset
DFFPC	ATV5000 ATV5100	D Flip-Flop with Asynchronous Preset and Reset
DFFSP	All	D Flip-Flop with Synchronous Preset
DFFSPC	All	D Flip-Flop with Synchronous Preset and Ssynchronous Reset. In the ATV750 and ATV2500 devices, the synchronous preset is a global preset product term, i.e., a single product term presets a group of flip-flops. In the ATV5000/ATV5100 devices, the preset term can be independently controlled.
JKFF	All	JK Flip-Flop with no Preset or Reset
JKFFC	All	JK Flip-Flop with Asynchronous Reset
JKFFP	ATV5000 ATV5100	JK Flip-Flop with Asynchronous Preset
JKFFPC	ATV5000 ATV5100	JK Flip-Flop with Asynchronous Preset and Reset
JFFSP	All	JK Flip-Flop with Synchronous Preset

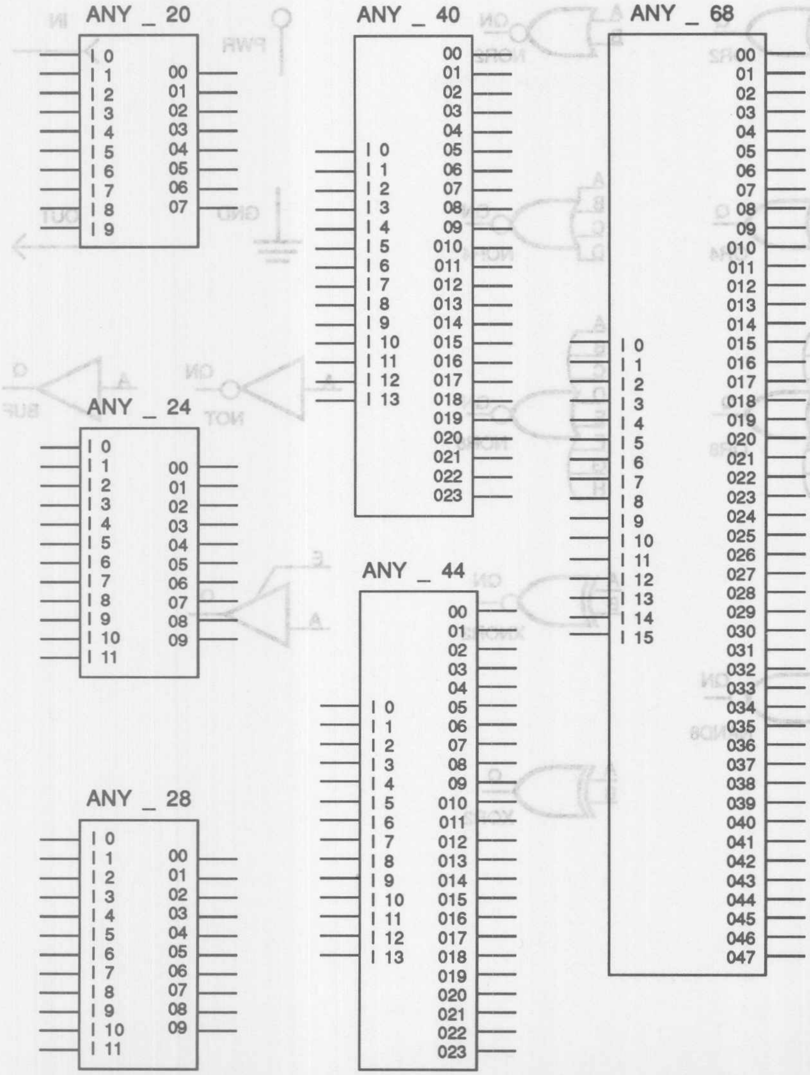
5

Table 2. Atmel-ViewPLD DIO Primitive Symbols (continued)

Primitive	Device Supported	Description
JKFFSPC	All	JK Flip-Flop with Synchronous Preset and Asynchronous Reset. In the ATV750 and ATV2500 devices, the synchronous preset is a global preset product term, i.e., a single product term presets a group of flip-flops. In the ATV5000/ATV5100 devices, the preset term can be independently controlled.
TFF	All	T Flip-Flop with no Preset or Reset
TFFC	All	T Flip-Flop with Asynchronous Reset
TFFP	ATV5000 ATV5100	T Flip-Flop with Asynchronous Preset
TFFPC	ATV5000 ATV5100	T Flip-Flop with Asynchronous Preset and Reset
TFFSP	ATV750 ATV2500	T Flip-Flop with Synchronous Preset
TFFSPC	ATV750 ATV2500	T Flip-Flop with Synchronous Preset and Asynchronous Reset. In the ATV750 and ATV2500 devices, the synchronous preset is a global preset product term, i.e., a single product term presets a group of flip-flops.
UDFDL	—	Viewlogic Built-In Primitive – DO NOT USE IN SCHEMATIC

Atmel's DIO Primitives, Sheet 1 of 4





ASHEET . 1
8.5 X 11"

ASHEET . 2
11 X 8.5"

BSHEET . 1
17 X 11"

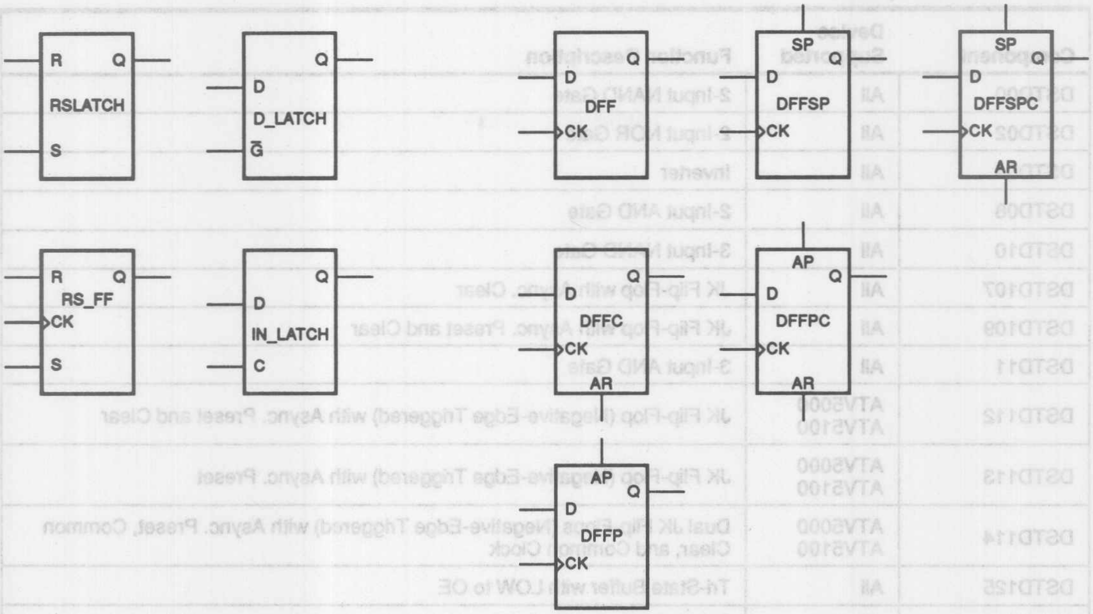
CSHEET . 1
17 X 22"

CSHEET . 2
22 X 17"

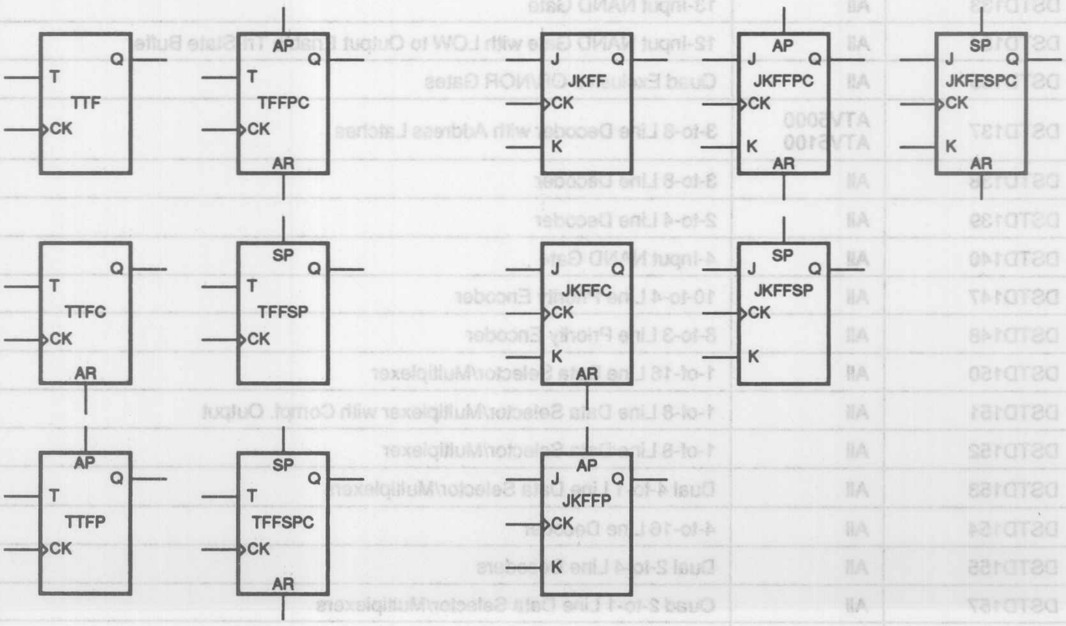
DSHEET . 1
34 X 22"

ESHEET . 1
34 X 44"

ESHEET . 2
44 X 34"



Atmel's DIO Primitives, Sheet 4 of 4



5

Table 3. DSTD TTL Symbols

Component	Device Supported	Function Description
DSTD00	All	2-Input NAND Gate
DSTD02	All	2-Input NOR Gate
DSTD04	All	Inverter
DSTD08	All	2-Input AND Gate
DSTD10	All	3-Input NAND Gate
DSTD107	All	JK Flip-Flop with Async. Clear
DSTD109	All	JK Flip-Flop with Async. Preset and Clear
DSTD11	All	3-Input AND Gate
DSTD112	ATV5000 ATV5100	JK Flip-Flop (Negative-Edge Triggered) with Async. Preset and Clear
DSTD113	ATV5000 ATV5100	JK Flip-Flop (Negative-Edge Triggered) with Async. Preset
DSTD114	ATV5000 ATV5100	Dual JK Flip-Flops (Negative-Edge Triggered) with Async. Preset, Common Clear, and Common Clock
DSTD125	All	Tri-State Buffer with LOW to OE
DSTD126	All	Tri-State Buffer with HIGH to OE
DSTD128	All	2-Input NOR Gate
DSTD133	All	13-Input NAND Gate
DSTD134	All	12-Input NAND Gate with LOW to Output Enable Tri-State Buffer
DSTD135	All	Quad Exclusive-OR/NOR Gates
DSTD137	ATV5000 ATV5100	3-to-8 Line Decoder with Address Latches
DSTD138	All	3-to-8 Line Decoder
DSTD139	All	2-to-4 Line Decoder
DSTD140	All	4-Input NAND Gate
DSTD147	All	10-to-4 Line Priority Encoder
DSTD148	All	8-to-3 Line Priority Encoder
DSTD150	All	1-of-16 Line Data Selector/Multiplexer
DSTD151	All	1-of-8 Line Data Selector/Multiplexer with Compl. Output
DSTD152	All	1-of-8 Line Data Selector/Multiplexer
DSTD153	All	Dual 4-to-1 Line Data Selector/Multiplexers
DSTD154	All	4-to-16 Line Decoder
DSTD155	All	Dual 2-to-4 Line Decoders
DSTD157	All	Quad 2-to-1 Line Data Selector/Multiplexers
DSTD158	All	Quad 2-to-1 Line Data Selector/Multiplexers with Compl. Output

Table 3. DSTD TTL Symbols (continued)

Component	Device Supported	Function Description
DSTD160	All	Sync. 4-Bit Decade Counter with Async. Clear
DSTD161	All	Sync. 4-Bit Binary Counter with Async. Clear
DSTD162	All	Sync. 4-Bit Decade Counter with Sync. Clear
DSTD163	All	Sync. 4-Bit Binary Counter with Sync. Clear
DSTD164	ATV5000 ATV5100	8-Bit Serial In/Parallel Out Shift Register with Async. Clear
DSTD165	ATV5000 ATV5100	8-Bit Parallel In/Serial Out Shift Register with Compl. Output
DSTD168	All	Synchronous 4-Bit Up/Down Decade Counter
DSTD169	All	Synchronous 4-Bit Up/Down Binary Counter
DSTD174	All	Hex D Flip-Flops with Async. Clear
DSTD175	All	Quad D Flip-Flops with Async. Clear and Compl. Output
DSTD176	ATV5000 ATV5100	4-Bit Presetable Decade Counter
DSTD177	ATV5000 ATV5100	4-Bit Presetable Binary Counter
DSTD180	All	9-Bit Parity Generator/Checker
DSTD181	ATV5000 ATV5100	Arithmetic Logic Unit/Function Generator
DSTD182	All	Look-Ahead Carry Generator
DSTD183	All	Carry-Save Full Address
DSTD190	ATV5000 ATV5100	Sync. 4-Bit Up/Down Decade Counter with Mode Control
DSTD191	ATV5000 ATV5100	Synchronous 4-Bit Up/Down Binary Counter with Mode Control
DSTD192	ATV5000 ATV5100	Synchronous 4-Bit Up/Down Decade Counter with Dual Clock
DSTD194	All	4-Bit Bidirectional Universal Shift Register
DSTD196	ATV5000 ATV5100	4-Bit Presettable Decade Counter
DSTD197	ATV5000 ATV5100	4-Bit Presettable Binary Counter
DSTD198	All	8-Bit Bidirectional Universal Shift Register
DSTD20	All	4-Input NAND Gate
DSTD21	All	4-Input AND Gate
DSTD226	ATV5000 ATV5100	4-Bit Parallel Latched Bus Tranceivers

5

Table 3. DSTD TTL Symbols (continued)

Component	Device Supported	Function Description
DSTD23	All	Dual 4-Input NOR Gates with Strobe
DSTD237	All	3-to-8 Line Decoder with Address Latches
DSTD240	All	Quad Tri-State Inverting Buffers
DSTD241	All	Octal Tri-State Buffers
DSTD242	All	Quad Tri-State Inverting Bus Transceivers
DSTD243	All	Quad Tri-State Bus Transceivers
DSTD244	All	Octal Tri-State Buffers
DSTD245	All	Octal Tri-State Bus Transceivers
DSTD25	All	4-Input NOR Gate with Strobe
DSTD251	All	Tri-State 1-to-8 Line Data Selector/Multiplexer with Compl. Output
DSTD253	All	Dual Tri-State 1-to-4 Multiplexers
DSTD257	All	Quad Tri-State 2-to-1 Multiplexers
DSTD258	All	Quad Tri-State 2-to-1 Inverting Multiplexers
DSTD260	All	5-Input NOR Gate
DSTD27	All	3-Input NOR Gate
DSTD273	All	Octal D Flip-Flops with Clear and Buffer Outputs
DSTD279	All	Dual SR Latches
DSTD28	All	2-Input NOR Buffer
DSTD280	All	9-Bit Odd/Even Parity Generators/Checkers
DSTD283	All	4-Bit Binary Full Adders with Fast Carry
DSTD290	All	Decade Counter
DSTD293	All	4-Bit Binary Counter
DSTD295	All	4-Bit Right/Left Shift Registers with Tri-State Outputs
DSTD298	All	Quad 2-Input Multiplexers with Storage
DSTD299	All	8-Bit Universal Shift/Storage Registers
DSTD30	All	8-Input NAND Gate
DSTD31	All	Delay Elements
DSTD32	All	2-Input OR Gate
DSTD323	All	8-Bit Universal Shift/Storage Registers
DSTD348	All	8-to-3 Line Priority Encoders with Tri-State Outputs
DSTD353	All	Dual 4-to-1 Line Data Selectors/Multiplexers with Tri-State Outputs
DSTD354	ATV5000 ATV5100	8-to-1 Multiplexer/Register (Transparent)

Table 3. DSTD TTL Symbols (continued)

Component	Device Supported	Function Description
DSTD356	All	8-to-1 Multiplexer/Register (Edge-Triggered)
DSTD365	All	Hex Bus Drivers with Tri-State Outputs
DSTD366	All	Hex Inverting Bus Drivers with Tri-State Outputs
DSTD367	All	Hex Bus Drivers with Tri-State Outputs with Two Tri-State Controls
DSTD368	All	Hex Inverting Bus Drivers with Tri-State Outputs with Two Tri-State Controls
DSTD37	All	2-Input NAND Buffers
DSTD373	ATV5000 ATV5100	Octal Transparent D Latches
DSTD374	All	Octal Edge-Triggered D Flip-Flops
DSTD376	All	Quad JK Flip-Flops
DSTD377	All	Octal D Flip-Flops with Enable
DSTD378	All	Hex D Flip-Flops with Enable
DSTD379	All	Quad D Flip-Flops with Enable
DSTD390	All	4-Bit Decade Counter
DSTD393	All	4-Bit Binary Counter
DSTD398	All	Quad 2-Input Multiplexer with Storage
DSTD399	All	Quad 2-Input Multiplexer with Storage
DSTD40	All	4-Input NAND Buffers
DSTD4002	All	4-Input NOR Gate
DSTD4016	All	Bilateral Switch
DSTD4017	All	Decade Counter/Divider
DSTD4020	All	14-Stage Ripple-Carry Binary Counter/Divider
DSTD4024	All	7-Stage Ripple-Carry Binary Counter/Divider
DSTD4040	All	12-Stage Ripple-Carry Binary Counter/Divider
DSTD4049	All	Inverting Buffer
DSTD4066	All	Bilateral Switch
DSTD4075	All	3-Input OR Gate
DSTD4078	All	8-Input NOR Gate with Compl. Output
DSTD42	All	BCD to Decimal Decoder
DSTD4514	ATV5000 ATV5100	8-Bit Static Shift Register
DSTD51	All	Dual 2-Wide 2-Input AND-OR-INVERT Gates
DSTD520	All	Octal 8-Bit Identity Comparator

Table 3. DSTD TTL Symbols (continued)

Component	Device Supported	Function Description
DSTD521	All	Octal 8-Bit Identity Comparator
DSTD533	ATV5000 ATV5100	Octal Tri-State Inverting Transparent D Latches
DSTD534	All	Octal Tri-State Inverting Edge-Triggered D Flip-flops
DSTD538	All	3-to-8 Line Decoders with Tri-State Outputs
DSTD54	All	4-Wide 2-Input AND-OR-INVERT Gates
DSTD540	All	Octal Buffers with Tri-State Outputs
DSTD541	All	Octal Inverting Buffers with Tri-State Outputs
DSTD55	All	2-Wide 4-Input AND-OR-INVERT Gates
DSTD563	ATV5000 ATV5100	Octal Tri-State Inverting Transparent D Latches
DSTD564	All	Octal Tri-State Inverting Edge-Triggered D Flip-Flops
DSTD568	All	Synchronous 4-Bit Up/Down Decade Counter with Tri-State Outputs
DSTD569	All	Synchronous 4-Bit Up/Down Binary Counter with Tri-State Outputs
DSTD573	ATV5000 ATV5100	Octal Tri-State Transparent D Latches
DSTD574	All	Octal Tri-State Edge-Triggered D Flip-Flops
DSTD575	All	Octal Tri-State Edge-Triggered D Flip-Flops with Sync. Clear
DSTD576	All	Octal Tri-State Inverting Edge-Triggered D Flip-Flops
DSTD580	ATV5000 ATV5100	Octal Tri-State Inverting Transparent D Latches
DSTD623	All	Octal Tri-State Bus Transceivers
DSTD638	All	Octal Inverting Bus Transceivers with Tri-State and Open-Collector Outputs
DSTD64	All	4-2-3-2-Input AND-OR-INVERT Gates
DSTD640	All	Octal Tri-State Inverting Bus Transceivers
DSTD643	All	Octal Tri-State True and Inverting Bus Transceivers
DSTD645	All	Octal Tri-State Bus Transceivers
DSTD646	All	Octal Bus Transceivers and Registers with AllTri-State Outputs
DSTD652	All	Octal Bus Transceivers and Registers
DSTD668	All	Synchronous 4-Bit Up/Down Decade Counters
DSTD669	All	Synchronous 4-Bit Up/Down Binary Counters
DSTD670	ATV5000 ATV5100	4-by-4 Register Files with Tri-State Outputs
DSTD677	All	16-to-4 Bit Address Comparator with Enable
DSTD684	All	8-Bit Magnitude/Identity Comparator

Table 3. DSTD TTL Symbols (continued)

Component	Device Supported	Function Description
DSTD688	All	8-Bit Identity Comparator
DSTD70	ATV5000 ATV5100	AND-Gated Edge-Triggered JK Flip-Flop with Async. Preset and Clear
DSTD73	ATV5000 ATV5100	JK Flip-Flop with Async. Clear
DSTD74	ATV5000 ATV5100	D Flip-Flop (Edge-Triggered) with Async. Preset and Clear
DSTD75	ATV5000 ATV5100	4-Bit Bistable Latches
DSTD76	ATV5000 ATV5100	JK Flip-Flop with Async. Preset and Clear
DSTD77	ATV5000 ATV5100	Dual Transparent D Latches
DSTD78	ATV5000 ATV5100	Dual JK Flip-Flops (Edge-Triggered) with Async. Preset, Common Clear, and Common Clock
DSTD80	All	Gated Full Adders
DSTD804	All	2-Input NAND Line Driver
DSTD805	All	2-Input NOR Line Driver
DSTD808	All	2-Input AND Line Driver
DSTD82	All	2-Bit Binary Full Adders
DSTD83	All	4-Bit Binary Full Adders with Fast Carry
DSTD832	All	2-Input OR Line Driver
DSTD85	All	4-Bit Magnitude Comparators
DSTD857	All	Hex 2-to-1 Universal Multiplexers
DSTD86	All	2-Input Exclusive-OR Gate
DSTD867	All	Synchronous 8-Bit Up/Down Counter with Async. Clear
DSTD869	All	Synchronous 8-Bit Up/Down Counter with Sync. Clear
DSTD874	All	4-Bit Edge-Triggered D Flip-Flop
DSTD878	All	4-Bit Edge-Triggered D Flip-Flop with Tri-State Outputs
DSTD882	All	32-Bit Look-Ahead Carry Generators
DSTD90	All	Decade Counter
DSTD91	All	8-Bit Shift Registers
DSTD92	All	Divide-by-12 Counter
DSTD93	All	4-Bit Binary Counter
DSTD94	ATV5000 ATV5100	4-Bit Shift Registers

5

Table 3. DSTD TTL Symbols (continued)

Component	Device Supported	Function Description
DSTD96	ATV5000 ATV5100	5-Bit Shift Registers
DSTDK	All	2-Input NAND Gate
DSTDK2	All	2-Input NOR Gate
DSTDK4	All	Inverter
DSTDK8	All	2-Input AND Gate
DSTD34	All	Buffer
DSTDK34	All	Buffer

Glossary

Term	Description
Async.	Asynchronous
Sync.	Synchronous
Compl.	Complementary

Using the ATV5000/ATV5100 Atmel-Abel™ Fitter

ATV5000 ATV5100 Atmel-Abel Fitter

Introduction

The ATV5000/ATV5100 Atmel-Abel Fitter is an automatic pin and node assignment program specifically written for ATV5000/ATV5100 designs implemented in the Abel Hardware Design Language (Abel HDL). The fitter utilizes the ATV5000 or ATV5100 device architecture resources to perform partial or complete pin and node signal assignments. For many designs, the fitter automatically assigns the pins and nodes efficiently without any manipulations by the designer. However, for very complex designs, some "steering" by the designer may be necessary in order to achieve high efficiency and performance.

This application note offers some hints that enable the ATV5000/ATV5100 fitter to achieve a higher design efficiency and performance. To maximize your understanding of these hints, you should read the Atmel ATV5000/ATV5100 Device Fitter manual. This manual contains a brief overview of the ATV5000 and ATV5100 architectures, the fitting process and fitter command options.

Interpreting the Fitter Log File (.FIT)

The ATV5000/ATV5100 fitter writes the pin and node assignments, device utilization, and any fitting errors the fitter encountered to the log file with the file extension `.FIT`. This fitter log file can be accessed via the `Fitter Assignments` command in the `View` menu window in the Abel Design Environment. Figure 1 shows an example of a fitter log file.

```
 Atmel P5000/P5100 fitter run on  4-Feb-94 10:25 AM.
======================================================
Warnings
======================================================
Fitting Device P5000.
Fitter mode=KEEP
Converting cluster toggle pin feedback to registered feedback.
======================================================
 The Design Fits.
======================================================
Overall Summary.
 Type       Used  Free   Utilization
____
Cell Usage
  Macrocells  2.25  72.75  4.3% (100 Wasted)
Pins Usage
  Inputs      0     8      0.0%
  Outputs     --    --     N/A
  IO          6     46     11.5%
  Summary     6     54     10.0%
======================================================
 Chip assignments summary.
Quadrant(1).
 toggle on 761
```

Figure 1. The `UR_RU.FIT` file for the ATV500 device (`KEEP` fitter option) *(continues)*

```
  RESET on 5
  clock_pin on 6
  input_pin on 7
  tog1 on 8
  toggle_RU_2 on 69
Quadrant(2).
  tog2 on 134
  toggle_UR_1 on 179
  tog3 on 18
Quadrant(3).
Quadrant(4).
========================================
  Quadrant by Quadrant Macrocell Usage.
.
.
.Node 196 (Buried_node) -> <none>
  (has_A has_upper_reg)
========================================
$DEVICE P5000 fit ur ru.tt3
$PINS 4 clock_pin:6 input_pin:5 tog1:7 tog3:18
$NODES 3 toggle:122 tog2:123 toggle_RU_1:72
```

Figure 1. *(continued)* The `UR_RU.FIT` file for the ATV500 device (KEEP fitter option)

When reviewing the fitter log file, you should pay special attention to the Chip Assignments Summary because it contains the information on how the fitter implemented the pin and node assignments. In addition to the pin and node assignments, the fitter may generate additional combinatorial nodes to route regional signals from one quadrant to another. The additional nodes generated contain suffixes such as _UR_n, _RU_n, and _n, where n is the total number of occurrences of the nodes.

Suffix	Description
_UR_n	Universal-to-Regional (UR) converters. These converters are used to convert the universal signals to regional signals. These conversions are sometimes necessary because of the limited number of universal product terms. Only the Buried Cells can be used for the UR conversion. This means that you can have up to a maximum of six UR converters in a quadrant of the ATV5000 or ATV5100 device.
_RU_n	Regional-to-Universal (RU) converters. These converters are used to route regional signals to other quadrants. The sum-term-feedbacks or B Nodes (nodes 69 to 120) are used for the RU conversion. Since the sum-term-feedback shares the same feedback path as the I/O pin, the sum-term-feedback is only available if the I/O pin is not used as an input, bidirectional I/O, combinatorial output, or a registered output with more than four product terms (or more than one universal product term for the ATV5000 device and two universal product terms for the ATV5100 device).
_n	These combinatorial nodes function similarly to the UR converters. They are generated for the ATV5100 regional control terms such as the Asynchronous Reset (AR) and Asynchronous Preset (AP) terms of the registers in the macrocells, and the Output Enable (OE) terms in the I/O cells. Like the UR converters, only the Buried Cells can be used for converting the universal signals to regional signals.

1. Universal signals are signals originating from the Universal Bus. Universal signals include signals from the I/O pins or sum-term-feedbacks (combinatorial nodes 69 to 120).
2. Regional signals are signals originating from the Regional Bus of quadrants 1, 2, 3, and 4. Regional signals include signals from the eight dedicated Clock/Latch Enable pins (also can be used as inputs), the buried Q1 and Q2 registers, and the Buried Cells.
3. Universal product terms are the product terms that are connected to both the Universal and Regional Buses.
4. Regional product terms are the product terms that are connected only to the Regional Bus in a quadrant.

```
"Inputs
RESET      pin 5;      "Reset for everybody
clock_pin  pin 6;      "Clock for everybody

"Quadrant 1
toggle  node 761  istype 'reg_t,buffer';        "A_node of pin 4
input_pin  pin 7       "Input to Universal bus
togl       pin 8       istype 'reg_t,buffer';

"Quadrant 2
tog2        node 134   istype 'reg_t,buffer';
tog3     pin 18        istype 'reg_t,buffer';

equations
"Quadrant 1 node 'toggle' wiggles when 'input_pin' is 1
toggle.t = input_pin;
toggle.clk = clock_pin;
toggle.ar = RESET;

"Quadrant 1 pin 'togl' wiggles when toggle is 1
togl.t = toggle;
togl.clk = clock_pin;
togl.ar = RESET;

"Quadrant 2 node 'tog2' wiggles when 'toggle' or 'togl' is 1
tog2.t = toggle # togl;
tog2.clk = clock_pin;
tog2.ar = RESET;

"Quadrant 2 pin 'tog3' wiggles when 'togl' and 'tog2' are 1
tog3.t = togl & tog2;
tog3.clk = clock_pin;
tog3.ar = RESET;
```

Figure 2. The `UR_RU.ABL` Abel file

In the `UR_RU.FIT` fitter log file in Figure 1, `toggle_RU_2` and `toggle_UR_1` are Regional-to-Universal (RU) and Universal-to-Regional (UR) converters. This is because the "toggle" signal is accessed by both the `tog1` and `tog2` output pins (see Figure 2 for the equations).

Note that the fitter option `KEEP` was used for the `UR_RU` design to illustrate the generation of the UR and RU converters. If there are no pin or node pre-assignments or if the `IGNORE` fitter option was used instead, then the fitter will make more efficient pin and node assignments.

Please refer to your Atmel ATV5000/ATV5100 Device Fitter manual for more information on the fitter log file.

Fitter Hints

The fitter hints described in this section help you fit your designs into an ATV5000 or ATV5100 device more efficiently. To implement these fitter hints, some modifications to your Abel design files may be necessary.

If you are an Atmel-ViewPLD user, you can modify your schematics to bring about similar Abel design modifications. If pin or node pre-assignments are recommended, they can be specified in the *design*`.PN` file. For more information on the `.PN` file, refer to the `ATMPIN.DOC` file. (The `ATMPIN` feature is supported in Atmel-ViewPLD Version 4.42 or higher).

Fitter Hint 1: Let the fitter makes its own pin and node assignments (IGNORE fitter option).

To allow maximum fitting efficiency, minimize the number of pre-assigned pins and nodes. With fewer constraints, the fitter can use its own pin and node assignments to its maximum capability. If you require pin pre-assignments to keep the fitter from changing the assignments (using the fitter option `KEEP`), then minimize the node pre-assignments. Note that in some designs, some pre-assignments may be able to "steer" the fitter to form or place signal groups that will lead to a higher fitting efficiency.

Figure 3 shows the `UR_RU` fitter log file with the `IGNORE` fitter option (ignore all pin and node pre-assignments). By ignoring the pre-assignments, the fitter was able to eliminate the use of the UR and RU converters, producing a more efficient and higher performance design fit.

```
Atmel P5000/P5100 fitter run on  4-Feb-94 03:28 PM.
==========================================
Warnings
==========================================
Fitting Device P5000.
Fitter mode=IGNORE
Converting cluster tog1 pin feedback to
registered feedback.
Converting cluster toggle pin feedback to registered feedback.
Converting cluster tog2 pin feedback to
registered feedback.
The Localized inputs are:
 RESET clock_pin input_pin
==========================================
 The Design Fits.
==========================================
  Overall Summary.
  Type  Used    Free   Utilization
 Cell Usage
  Macrocells    1.00   75.00     1.3%  (0.00
                                 Wasted)
 Pins Usage
  Inputs     3 5        37.5%
  Outputs   --          --     N/A
  IO     2  50   3.8%
  Summary  5   55      8.3%
==========================================
 Chip assignments summary.
Quadrant(1).
 clock_pin on 1
 RESET on 2
 tog3 on 15
 tog1 on 17
 tog2 on 133
 toggle on 132
Quadrant(2).
 input_pin on 32
Quadrant(3).
Quadrant(4).
```

Figure 3. The `UR_RU.FIT` file for the ATV5000 device (`IGNORE` fitter option)

Fitter Hint 2: Use the pin clock feature to break up large signal groups.

Atmel-ViewPLD Users: Use the symbols from the PDFF or CDFF primitive family in the DIO library. The PDFF primitives implement the pin clocking features, whereas the CDFF primitives allow the gated pin clocking functions.

If you have a large signal group (a function group that cannot fit into a single quadrant), then using the pin clocking or clock-enable feature (that is, using `signal.CE = pin_clk` equations) may help the fitter break up the large signal group into smaller, more efficient groups. Figures 4 and 5 show the `CNT10.ABL` 10-bit counter equations utilizing the product term and pin clocking features, respectively.

```
CNTA.T = ((CNTA.FB + 1) $ CNTA.FB) & CE   "CNTA = [A9..A0], Counter A
 # (INPUT $ CNTA.FB) & LD;                 "INPUT = [I9..I0]
CNTA.C = CLK1;                             "Independent product term
                                           "clocking
CNTA.AR = RST;
```

Figure 4. `CNT10.ABL` counter equations with product term clocking

```
CNTA.T = ((CNTA.FB + 1) $ CNTA.FB) & CE   "CNTA = [A9..A0], Counter A
 # (INPUT $ CNTA.FB) & LD;                 "INPUT = [I9..I0]
CNTA.CE = CLK1;   CNTA.CK = ^h3FF;         ".CE equation to
                                           "implement pin clocking
CNTA.AR = RST;
```

Figure 5. `CNT10.ABL` counter equations with pin clocking

Figures 6 and 7 show the chip assignments summary of the CNT10 fitter log files for the designs with product term and pin clocking options. The CNT10 design consists of four 10-bit loadable counters in which the outputs of one counter are loaded into the next counter. Since the outputs of the counters are inputs for the next counters, the fitter will form a single large group for these counters. As illustrated by the fitter log files, the fitter separated each counter more efficiently in the design with the pin clocking option. In Figure 6, some of the counter B outputs were scattered to other quadrants instead of being

```
 Chip assignments summary.
Quadrant(1).
 CE on 1
 LD on 2
 CLK on 4      <clock for counters A, B, C and D via product terms>
 RST on 5
 I6 on 6
 A9 on 7
 A8 on 8
 A7 on 9
 A6 on 10
 A5 on 11
 A4 on 12
 A3 on 13
 A2 on 14
 A1 on 15
 A0 on 17
Quadrant(2).
 I7 on 18
 I8 on 19
 C9 on 21
 C8 on 22
 C7 on 23
 C6 on 24
 C5 on 25
 C4 on 26
 C3 on 27
 C2 on 28
 C1 on 29
 C0 on 30
 B0 on 31      <Counter B output in Quadrant 2>
    I0 on 32
    I1 on 34
Quadrant(3).
    I3 on 35
    I2 on 36
    I9 on 38
    D9 on 40
    D8 on 41
    D7 on 42
    D6 on 43
    D5 on 44
    D4 on 45
    D3 on 46
    D2 on 47
    D1 on 48
    D0 on 49
    B1 on 51   <Counter B output in Quadrant 3>
Quadrant(4).
    B9 on 58
    B8 on 59
    B7 on 60
    B6 on 61
    B5 on 62
    B4 on 63
    B3 on 64
    B2 on 65
    I4 on 66    I5 on 68
```

Figure 6. Fitter log file for CNT10.ABL with product term clocking

5

grouped together in a single quadrant. For this design example, these scattered placements of the counter B outputs pose no performance penalty. But for a very complex design, the scattered pin or node placements may inhibit efficient fitting.

```
Chip assignments summary.
Quadrant(1).
    LD on 1
    CLK1 on 2            (clock for counter A)
    RST on 4
    I2 on 5
    I3 on 6
    A9 on 7
    A8 on 8
    A7 on 9
    A6 on 10
    A5 on 11
    A4 on 12
    A3 on 13
    A2 on 14
    A1 on 15
    A0 on 17
Quadrant(2).
    I4 on 18
    I5 on 19
    I6 on 21
    B9 on 22
    B8 on 23
    B7 on 24
    B6 on 25
    B5 on 26
    B4 on 27
    B3 on 28
    B2 on 29
    B1 on 30
    B0 on 31
    CLK2 on 32           (clock for counter B)
    CE on 34
Quadrant(3).
    I0 on 35
    CLK3 on 36
    I7 on 38
    I8 on 39
    I9 on 40
    C9 on 41
    C8 on 42
    C7 on 43
    C6 on 44
    C5 on 45
    C4 on 46
    C3 on 47
    C2 on 48
    C1 on 49
    C0 on 51             (clock for counter C)
Quadrant(4).
    D9 on 56
    D8 on 57
    D7 on 58
    D6 on 59
    D5 on 60
    D4 on 61
    D3 on 62
    D2 on 63
    D1 on 64
    D0 on 65
    CLK4 on 66           (clock for counter D)
    I1 on 68
```

Figure 7. Fitter log file for `CNT10.ABL` with pin clocking

Fitter Hint 3: Reduce the number of RU converters by promoting registered nodes that are referenced by other pins or nodes placed in other quadrants to pins

Atmel-ViewPLD Users: Add and connect the OUT symbol from the DIO library to the registered nodes.

The ATV5000/ATV5100 automatically generates RU converters so that the regional signals from one quadrant can be routed to other quadrants. In the fitter log files, the RU converters are indicated by the suffix `_RU_n`, where `n` is the total number of occurrences of the RU and UR converters. In Figure 8, the chip assignments summary of the JACK5000 fitter log file (`JACK5000.FIT`) shows that RU converters were generated

```
 Atmel P5000/P5100 fitter run on  7-Feb-94 10:41 PM.
==========================================
Warnings
==========================================
Fitting Device P5000.
Fitter mode=IGNORE
Converting cluster Ace pin feedback to registered feedback.
Converting cluster S3 pin feedback to registered feedback.
Converting cluster S2 pin feedback to registered feedback.
Converting cluster S1 pin feedback to registered feedback.
Converting cluster S0 pin feedback to registered feedback.
The Localized inputs are:
 V4 V3 V2 V1 V0
Universal element Add10 regionalized.
Universal element Sub10 regionalized.
Universal element Q0 regionalized.
Universal element Q2 regionalized.
Universal element Add10 regionalized.
Universal element Sub10 regionalized.
Signal C2 inverted to get better term usage.
==========================================
 The Design does NOT fit.
==========================================
  Overall Summary.
  Type     Used      Free   Utilization
 ________________________
 Cell Usage
  Macrocells  18.75  53.50  29.4% (3.50 Wasted)
 Pins Usage
  Inputs      5      3      62.5%
  Outputs     --     --     N/A
  IO          15     37     28.8%
  Summary     20     40     33.3%
==========================================
Chip assignments summary.
Quadrant(1).
    V3 on 1
    V4 on 2
    S1 on 761
    S3 on 762
    S2 on 763
    Clk on 7
    D5 on 8
    D3 on 9
    D2 on 10
    D4 on 11
    D1 on 12
    Q2_UR_4 on 78
    C4 on 79
    Bust on 80
    Hit on 81
    Add10_UR_1 on 173
    C3 on 174
    Sub10_UR_2 on 175
    C2 on 176
    C1 on 177
    Q0_UR_3 on 178
```

Figure 8. Original chip assignments summary of the `JACK5000.FIT` file *(continues)*

```
    CardOut on 6
  1 partitioned but unassigned signals.
 Q1
Quadrant(2).
    Sub10 on 27
    Add10 on 28
    Q2 on 29
    Q0 on 30
    Ace on 31
    V2 on 32
    V1 on 34
    is_Ace on 179
Quadrant(3).
    V0 on 36
    D0 on 49
    S0_RU_7 on 107    <RU Converter>
 Converter
    Add10_UR_5 on 185
    Sub10_UR_6 on 186
    S0 on 159
Quadrant(4).
    S4_RU_8 on 120    <RU Converter>
 Converter
  1 partitioned but unassigned signals.
 S4
```

Figure 8. *(Continued)* Original chip assignments summary of the `JACK5000.FIT` file

for the S0 and S4 registered nodes. The RU converters were generated because these regional registered nodes are being referenced by the nodes placed in other quadrants. A look at the optimized equations shown in Figure 9 confirms that the S0 and S4 nodes are inputs to the D0, D1, D2, D3 and D4 output pins.

```
D0 = (S0.FB);

D1 = (!S4.FB & !S3.FB & S1.FB
     # !S3.FB & S2.FB & S1.FB
     # S4.FB & S3.FB & !S2.FB & S1.FB
     # !S4.FB & S3.FB & S2.FB & !S1.FB
     # S4.FB & !S3.FB & !S2.FB & !S1.FB);

D2 = (!S4.FB & !S3.FB & S2.FB
     # S4.FB & S3.FB & !S2.FB
     # !S4.FB & S2.FB & S1.FB
     # S4.FB & !S2.FB & !S1.FB);

D3 = (S4.FB & !S3.FB & !S2.FB & S1.FB
     # S4.FB & S3.FB & S2.FB & !S1.FB
     # !S4.FB & S3.FB & !S2.FB & !S1.FB);

D4 = (!S4.FB & S3.FB & S2.FB
     # S4.FB & !S3.FB & !S2.FB
     # !S4.FB & S3.FB & S1.FB
     # S3.FB & S2.FB & S1.FB);

D5 = (S4.FB & S3.FB
     # S4.FB & S2.FB);
```

Figure 9. Optimized equations of JACK5000 design example

After promoting the S0 and S4 nodes to pins, the JACK5000 Abel file was re-compiled. As shown in Figure 10, the updated fitter log file now contains a few more RU converters. The nodes promotion process is repeated for the S1, S2, and S3 registered nodes.

```
 Atmel P5000/P5100 fitter run on  7-Feb-94 10:39 PM.
==========================================
Warnings
==========================================
Fitting Device P5000.
Fitter mode=IGNORE
Converting cluster Ace pin feedback to registered feedback.
Converting cluster S0 pin feedback to registered feedback.
Converting cluster S3 pin feedback to registered feedback.
Converting cluster S2 pin feedback to registered feedback.
Converting cluster S1 pin feedback to registered feedback.
The Localized inputs are:
 V4 V3 V2 V1 V0
Universal element Add10 regionalized.
Universal element Sub10 regionalized.
Universal element Q0 regionalized.
Universal element Q2 regionalized.
Signal C2 inverted to get better term usage.
==========================================
 The Design does NOT fit.
==========================================
 Overall Summary.
 Type        Used   Free      Utilization
 Cell Usage
  Macrocells  17.50  54.25     28.4%  (4.00 Wasted)
 Pins Usage
  Inputs      5      3         62.5%
  Outputs     --     --        N/A
  IO          16     36        30.8%
  Summary     21     39        35.0%
==========================================
 Chip assignments summary.
Quadrant(1).
    V3 on 1
    V4 on 2
    S1 on 761
    S3 on 762
    S2 on 763
    Clk on 7
    D2 on 8
    D1 on 9
    S1_RU_7 on 75
 Converter
    S2_RU_6 on 76
 Converter
    S3_RU_5 on 77
 Converter
    Q2_UR_4 on 78
    C4 on 79
    Bust on 80
    Hit on 81
    Add10_UR_1 on 173
    C3 on 174
    Sub10_UR_2 on 175
    C2 on 176
    C1 on 177
    Q0_UR_3 on 178
    Clr on 4
    CardIn on 5
    CardOut on 6
```

Figure 10. Chip assignments summary of the JACK5000.FIT file after promoting S0 and S4 nodes to pins *(continues)*

```
  1 partitioned but unassigned signals.
 Q1
Quadrant(2).
    Sub10 on 27
    Add10 on 28
    Q2 on 29
   Q0 on 30
     Ace on 31
    V2 on 32
    V1 on 34
    is_Ace on 179
Quadrant(3).
    V0 on 36
    D0 on 49
    S0 on 51
  1 partitioned but unassigned signals.
  S4
Quadrant(4).
    D5 on 63
    D3 on 64
    D4 on 65
```

Figure 10. *(Continued)* Chip assignments summary of the JACK5000.FIT file after promoting S0 and S4 nodes to pins

With the promotion of S0, S1, S2, S3 and S4 registered nodes to pins, the JACK5000 design successfully fit into the ATV5000 device. The chip assignments summary in Figure 11 shows that all the RU converters were successfully eliminated.

```
 Atmel P5000/P5100 fitter run on  7-Feb-94 10:34 PM.
==========================================
Warnings
==========================================
Fitting Device P5000.
Fitter mode=IGNORE
Converting cluster Ace pin feedback to registered feedback.
Converting cluster S3 pin feedback to registered feedback.
Converting cluster S2 pin feedback to registered feedback.
Converting cluster S1 pin feedback to registered feedback.
Converting cluster S0 pin feedback to registered feedback.
The Localized inputs are:
 V4 V3 V2 V1 V0
Universal element Sub10 regionalized.
Universal element Add10 regionalized.
Universal element S4 regionalized.
Universal element S3 regionalized.
Universal element S3 regionalized.
Universal element S1 regionalized.
Universal element S4 regionalized.
Signal C2 inverted to get better term usage.
==========================================
 The Design Fits.
==========================================
  Overall Summary.
 Type         Used      Free      Utilization
________________________________
 Cell Usage
  Macrocells  22.00     51.25     32.6%  (2.75 Wasted)
 Pins Usage
  Inputs      5         3         62.5%
  Outputs     --        --        N/A
  IO          21        31        40.4%
  Summary     26        34        43.3%
==========================================
```

Figure 11. Chip assignments summary of the JACK5000.FIT file after promoting S0, S1, S2, S3, and S4 nodes to pins *(continues)*

```
 Chip assignments summary.
Quadrant(1).
    V3 on 1
    V4 on 2
    Clk on 4
    Clr on 5
    CardIn on 6
    CardOut on 7
    S2 on 13
    S3 on 14
    S4 on 15
    S1 on 17
    C4 on 173
    C3 on 174
    C2 on 175
    C1 on 176
    Sub10_UR_1 on 177
    Add10_UR_2 on 178
Quadrant(2).
    Add10 on 26
    Sub10 on 27
    Q2 on 28
    Ace on 29
    Q0 on 30
    Q1 on 31
    V2 on 32
    V1 on 34
    Hit on 179
    Bust on 180
    is_Ace on 181
    S4_UR_3 on 182
    S3_UR_4 on 183
Quadrant(3).
    V0 on 36
    D5 on 46
    D3 on 47
    D2 on 48
    D4 on 49
    D1 on 51
    S3_UR_5 on 185
    S1_UR_6 on 186
    S4_UR_7 on 187
Quadrant(4).
    D0 on 64
    S0 on 65
```

Figure 11. *(Continued)* Chip assignments summary of the `JACK5000.FIT` file after promoting S0, S1, S2, S3, and S4 nodes to pins

Fitter Hint 4: Reduce the number of RU and UR converters by targeting the ATV5100 instead of the ATV5000.

The ATV5100 device has an architecture very similar to the ATV5000. One advantage the ATV5100 has over the ATV5000 is that there are more universal routing resources for logic functions in the ATV5100 (a maximum of 8 universal product terms in the ATV5100 versus 4 in the ATV5000). This device is well suited for applications that have large signal groups (for example, large state machines) which can not be fitted into a single quadrant.

Figure 12 shows the ATV5100 chip assignments summary of the original JACK5000 Abel source file . Comparing the fitter log files in Figures 8 and 12, the design fits more efficiently in the ATV5100 device than the ATV5000 device.

```
 Atmel P5000/P5100 fitter run on  8-Feb-94 11:54 AM.
==========================================
Warnings
==========================================
Fitting Device P5100.
Fitter mode=IGNORE
Converting cluster Ace pin feedback to registered feedback
```

Figure 12. ATV5100 chip assignments summary of the original JACK5000 file *(continues)*

```
Converting cluster S2 pin feedback to registered feedback.
Converting cluster S1 pin feedback to registered feedback.
Converting cluster S0 pin feedback to registered feedback.
The Localized inputs are:
 V4 V3 V2 V1 Clr
Universal element Add10 regionalized.
Universal element Sub10 regionalized.
Universal element V0 regionalized.
Signal C2 inverted to get better term usage.
==========================================
 The Design Fits.
==========================================
  Overall Summary.
  Type          Used      Free      Utilization
 Cell Usage
  Macrocells    15.75     56.50     25.4%  (3.50 Wasted)
 Pins Usage
  Inputs        5         3         62.5%
  Outputs       --        --        N/A
  IO            16        36        30.8%
  Summary       21        39        35.0%
==========================================
 Chip assignments summary.
Quadrant(1).
    V3 on 1
    V4 on 2
    S1 on 761
    S3 on 762
    S2 on 763
    Clk on 7
    D5 on 8
    D3 on 9
    D2 on 10
    D4 on 11
    D1 on 12
    Q1 on 13
    C4 on 79
    Bust on 80
    Hit on 81
    C2 on 173
    C3 on 174
    C1 on 175
    Add10_UR_1 on 176
    Sub10_UR_2 on 177
    V0 on 4
    CardIn on 5
    CardOut on 6
Quadrant(2).
    Sub10 on 27
    Add10 on 28
    Q2 on 29
    Q0 on 30
    Ace on 31
    V2 on 32
    V1 on 34
    is_Ace on 179
Quadrant(3).
    Clr on 36
    D0 on 49
    S0_RU_4 on 107
    V0_UR_3 on 185
    S0 on 159
Quadrant(4).
    S4 on 800
    S4_RU_5 on 120
```

Figure 12. *(Continued)* **ATV5100 chip assignments summary of the original JACK5000 file**

By applying the fitter hint 3, the RU converters can be removed by promoting the S0, S1, S2, S3, and S4 registered nodes to pins. The fitter log file in Figure 13 shows that there are only two UR converters compared to six converters for the ATV5000 device (refer to Figure 11).

```
 Atmel P5000/P5100 fitter run on  8-Feb-94 01:09 PM.
===========================================
Warnings
===========================================
Fitting Device P5100.
Fitter modeIGNORE
Converting cluster Ace pin feedback to registered feedback.
Converting cluster Add10 pin feedback to registered feedback.
Converting cluster Sub10 pin feedback to registered feedback.
Converting cluster S4 pin feedback to registered feedback.
Converting cluster S3 pin feedback to registered feedback.
Converting cluster S2 pin feedback to registered feedback.
Converting cluster S1 pin feedback to registered feedback.
Converting cluster S0 pin feedback to registered feedback.
The Localized inputs are:
V4 V3 V2 V1 Clr
Universal element Add10 regionalized.
Universal element Sub10 regionalized.
Signal C2 inverted to get better term usage.
===========================================
 The Design Fits.
===========================================
  Overall Summary.
  Type          Used        Free       Utilization
_______________
 Cell Usage
  Macrocells  16.25       57.00        25.0%  (2.75 Wasted)
 Pins Usage
  Inputs      5           3            62.5%
  Outputs     --          --           N/A
  IO          21          31           40.4%
  Summary     26          34           43.3%
===========================================
 Chip assignments summary.
Quadrant(1).
    V3 on 1
    V4 on 2
    Clk on 4
    V0 on 5
    CardIn on 6
    CardOut on 7
    S2 on 13
    S3 on 14
    S4 on 15
    S1 on 17
    C4 on 173
    C3 on 174
    C2 on 175
    C1 on 176
    Add10_UR_1 on 177                   <1st UR converter>
    Sub10_UR_2 on 178                   <2nd UR converter>
Quadrant(2).
    Add10 on 26
    Sub10 on 27
    Q2 on 28
    Ace on 29
    Q0 on 30
    Q1 on 31
    V2 on 32
    V1 on 34
    Hit on 179
```

Figure 13. ATV5100 chip assignments summary of the `JACK5000.FIT` file after promoting S0, S1, S2, S3, and S4 nodes to pins *(continues)*

5

```
        Bust on 180
        is_Ace on 181
    Quadrant(3).
        Clr on 36
        D5 on 46
        D3 on 47
        D2 on 48
        D4 on 49
        D1 on 51
    Quadrant(4).
        D0 on 64
        S0 on 65
```

Figure 13. *(Continued)* ATV5100 chip assignments summary of the `JACK5000.FIT` file after promoting S0, S1, S2, S3, and S4 nodes to pins

Fitter Hint 5: Reduce the UR converters by promoting registered nodes with many product terms to pins.

Normally, the ATV5000/ATV5100 fitter will choose the Default or Reverse polarity equations with the fewest number of product terms for the combinatorial nodes and for both the registered and combinatorial output pins (refer to Figure 14). But for the registered nodes, the ATV5000/ATV5100 fitter only implements the Default polarity equations (inverting is not done because of complications arising from T-type flip-flops and reset and preset configurations).

The Abel Optimizer (PLAOPT) produces two sets of equations, Default and Reverse polarity, for each equation in your design.

Both equation sets are functionally equivalent. However, in order to select the Reverse polarity equations, the fitter must invert the polarity of the output signals. This inversion can be achieved simply by setting the polarity of the Atmel PLD outputs to negative or invert (all Atmel PLDs have polarity control on their outputs). For internal combinatorial nodes without polarity control capabilities, the ATV5000/ATV5100 fitter accomplishes the inversions by inverting the references to these nodes in all equations.

Figure 14 shows the FIT50A Abel example, which illustrates how to ensure that both the Default and Reverse polarity equations are accessible to the fitter.

```
module FIT50A;
title 'A simple design example that the fitter will choose the Default or
      Reverse polarity equations with the fewest number of product terms.';

"Use KEEP fitter option to force to I0..I7 to use the I/O pins as inputs.

CLK pin 1;
RST pin 2;
I0,I1,I2,I3,I4,I5,I6,I7 pin 18,19,21,22,23,24,25,26;

BurCOM node istype 'com';        "COM Node is OK because fitter
                                 "automatically choose polarity.  Note that if
                                 "the Reverse polarity is chosen, vectors may
                                 "failed due to the inversion (if this node is
                                 "simulated). In this case, simply add the ! to
                                 "the node label in the TEST_VECTORS
                                 "section would allow the node to be
                                 "simulated.
BurREG node istype 'reg_d';      "Should promote this node to pin because
                                 "Default polarity equations are used for
                                 "all the registered nodes.

OUTCOM pin istype 'com';         "Fitter will choose polarity equations.

OUTREG pin istype 'reg_d';       "Fitter will choose polarity equations as long
                                 "as the .D dot extensions are not used in the
                                 "equations. Do not use BUFFER as it will
                                 "force fitter to use only the Default polarity.
                                 "Default polarity will always be chosen for
                                 "REG_T type register.  If the Reverse polarity
                                 "is used, the Reset (power-up and
                                 "asynchonous) and Preset (asynchonous)
                                 "vectors may need to be changed.

"CREATE BUSES
Input = [I7..I0];
EQUATIONS
                                       "Default Polarity / Reverse Polarity
BurCOM  = !(Input == 1);               "  8 p-terms      /   1 p-term
OUTCOM  = !((Input == 1) & BurCOM);    "  9 p-terms      /   1 p-term

                                       "Default Polarity / Reverse Polarity
BurREG := !(Input == 1);               "  8 p-terms      /   1 p-term
BurREG.c = CLK;                        "If REG_D type flip-flop, use the :=
BurREG.ar = RST;                       "instead of the .D so that the fitter
                                       "or fusemapper will choose Default
                                       "or Reverse polarity equations with
                                       "the fewest product-terms

OUTREG := !((Input == 1) & BurREG);    "  9 p-terms      /   1 p-term
OUTREG.c = CLK;
OUTREG.ar = RST;

END;
```

Figure 14. FIT50A Abel design example to allow fitter to choose Default or Reverse polarity equations

5

The chip assignments summary in Figure 15 shows that there are four UR converters. A quick look at the FIT50A PLA optimized file (Figure 16) confirms that the Reverse polarity equation of the BurREG node should be used (that is, one product term versus eight product terms in the Default equation). Using fitter hint 5, these UR converters could be eliminated by promoting the BurREG node to pin.

```
 Atmel P5000/P5100 fitter run on  8-Feb-94 03:17 PM.
====================================================
Warnings
====================================================
Fitting Device P5000.
Fitter mode=KEEP
Converting cluster BurREG pin feedback to registered feedback.
Clust BurREG had 8 universals (possible 4) which needed patching.
Universal element I0 was regionalized.
Clust BurREG had 7 universals (possible 4) which needed patching.
Universal element I1 was regionalized.
Clust BurREG had 6 universals (possible 4) which needed patching.
Universal element I2 was regionalized.
Clust BurREG had 5 universals (possible 4) which needed patching.
Universal element I3 was regionalized.
Signal BurCOM inverted to get better term usage.    <Reverse polarity
                                                  equation used for Burcom>
====================================================
 The Design Fits.
====================================================
  Overall Summary.
  Type        Used    Free     Utilization
 ________________________________
 Cell Usage
  Macrocells  6.25    67.75    10.9%  (2.00 Wasted)
 Pins Usage
  Inputs      2       6        25.0%
  Outputs     --      --       N/A
  IO          10      42       19.2%
  Summary     12      48       20.0%
====================================================
 Chip assignments summary.
Quadrant(1).
    CLK on 1
    RST on 2
    BurREG on 761
    OUTREG on 15
    OUTCOM on 17
    BurCOM on 173
    I0_UR_1 on 174
    I1_UR_2 on 175
    I2_UR_3 on 176
    I3_UR_4 on 177
Quadrant(2).
    I0 on 18
    I1 on 19
    I2 on 21
    I3 on 22
    I4 on 23
    I5 on 24
    I6 on 25
    I7 on 26
Quadrant(3).
Quadrant(4).
```

Figure 15. Original FIT50A fitter log file (no modifications)

```
Product Term Usage:
          Default  Reverse   Signal
          Polarity Polarity  Signal
          _______  _______   ______
             8        1      BurCOM
             9        1      OUTCOM
             8        1      BurREG.REG
             1        1      BurREG.C
             1        1      BurREG.AR
             9        1      OUTREG.REG
             1        1      OUTREG.C
             1        1      OUTREG.AR
```

Figure 16. FIT50A PLA Optimized file

```
 Atmel P5000/P5100 fitter run on  8-Feb-94 04:03 PM.
================================================
Warnings
================================================
Fitting Device P5000.
Fitter mode=KEEP
Converting cluster BurREG pin feedback to registered feedback.
Signal BurCOM inverted to get better term usage.
==========================================
 The Design Fits.
==========================================
  Overall Summary.
  Type          Used    Free   Utilization
_______________
 Cell Usage
  Macrocells   1.75    72.25    4.9%  (2.00 Wasted)
 Pins Usage
  Inputs       2       6        25.0%
  Outputs      --      --       N/A
  IO           11      41       21.2%
  Summary      13      47       21.7%
==========================================
 Chip assignments summary.
Quadrant(1).
    CLK on 1
    RST on 2
    OUTREG on 14
    BurREG on 15
    OUTCOM on 17
    BurCOM on 173
Quadrant(2).
    I0 on 18
    I1 on 19
    I2 on 21
    I3 on 22
    I4 on 23
    I5 on 24
    I6 on 25
    I7 on 26
Quadrant(3).
Quadrant(4).
```

Figure 17. Fit50A fitter log file (with promotion of BurREG node to pin)

5

Fitter Hint 6: Use the fitter option NO_FB_CONVERT to reduce the UR converters.

In the default condition (that is, without the NO_FB_CONVERT option), the ATV5000/ATV5100 fitter will automatically convert references of all registered output-only pins (non-three-statable output pins) so that any equations referencing these signals will utilize their (regional) internal Q1 registered feedbacks (via the Abel .FB extension) instead of the (universal) pin feedbacks (that is, feedbacks directly from the output pins). Refer to the ATV5000 macrocell configuration in Figure 9 of the ATV5000 data sheet for more information on the feedback types. Which feedbacks you use is important because the internal Q1 registered feedbacks are regional signals, and the pin feedbacks are universal signals. With most designs, utilizing the internal Q1 registered feedbacks will result in the optimal fitting efficiency.

In the fitter log file of the Abel design example FIT50, the message *"Converting cluster A pin feedback to registered feedback"* means that the internal Q1 registered feedback of the output A will be used for all equations referencing it. The fitter appends the .FB extension to all references of the output A.

```
FIT50 Abel design example:
Original Abel equation: B.D = (I & A);

"A is registered output-only pin
Fitter output equation: B.D = (I & A.FB);
```

With the NO_FB_CONVERT option enabled (specify NO_FB_CONVERT in the Alternate Fitter Strategy box in the Fit Options... command window), the ATV5000/ATV5100 fitter disables the "registered feedback" conversions. This means that the fitter will not modify the references of the registered output-only pins and will use the .FB, .Q, or .PIN (or no extension) feedback types specified in the original Abel source file. By specifying the feedback types, you may be able to reduce the number of UR converters in your designs.

```
module FIT50;
title 'Fitter example showing the use of NO_FB_CONVERT fitter option
       to minimize the Universal-to-Regional (UR) converters for the
       ATV5000 or ATV5100';
"Use fitter option "KEEP" and "NO_FB_CONVERT"

CLK pin;
RST pin;
I pin;

A pin 4 istype 'reg_d,buffer';
B pin 5 istype 'reg_d,buffer';
C pin 6 istype 'reg_d,buffer';

D node 774 istype 'reg_d,buffer';
E pin 19 istype 'reg_d,buffer';

"Input Bus
Regs = [A,B,C,D,E];

EQUATIONS

Regs.c = CLK;
Regs.ar = RST;

A.d = I;               "Output A forced to Quadrant 1
B.d = I & A.FB;        "Output B forced to Quadrant 1
C.d =  A.FB & B.FB     "Output C forced to Quadrant 1
     # A.FB & C.FB     "Output C requires the .FB feedbacks to be used in
     # B.FB & C.FB     "some product terms because both A and B outputs are
     # B & E           "in the same quadrant as output C. Output E is in
     # C & E;          "quadrant 2 so the last two product terms will be
                       "universal regardless of whether .FB feedbacks are used
                       "for B and C feedbacks.

D.d = A & B            "Node D forced to Quadrant 2
     # A & C           "Keep feedbacks A, B and C as universal product signals
     # B & C;          "(maximum universal product terms for a registered node
                               "is 4).

E.d =  D;             "Output E forced to Quadrant 2

END;
```

Figure 18. FIT50 Abel design example for illustrating the NO_FB_CONVERT fitter option

Note: The NO_FB_CONVERT fitter option will only be effective when registered nodes in one quadrant reference the registered output-only pins located in other quadrants.

Figure 18 shows the Abel equations of the FIT50 design example. To illustrate the NO_FB_CONVERT option, the KEEP fitter option was used to force the fitter to place the output pins and nodes in different quadrants (directed by the pin and node pre-assignments).

```
 Chip assignments summary.
Quadrant(1).
    RST on 1
    CLK on 2
    A on 4
    B on 5
    C on 6
Quadrant(2).
    D on 774
    E on 19
    I on 32
    A_UR_1 on 179
    B_UR_2 on 180
    C_UR_3 on 181
Quadrant(3).
Quadrant(4).
```

Figure 19. Chip assignments summary of FIT50 without NO_FB_CONVERT fitter option

```
 Chip assignments summary.
Quadrant(1).
    RST on 1
    CLK on 2
    A on 4
    B on 5
    C on 6
Quadrant(2).
    D on 774
    E on 19
    I on 32
Quadrant(3).
Quadrant(4).
```

Figure 20. Chip assignments summary of FIT50 with NO_FB_CONVERT fitter option

5

Using the Programmable Polarity Control

The output programmable polarity control in PLDs brings efficiency in logic reduction and control of output polarity to the customers. Unfortunately, it also brings confusion to customers who are not familiar with the software syntax to properly configure the output polarity.

This application note shows the proper usage of the popular Abel™ and Cupl™ syntax to configure the output polarity of Atmel PLDs.

Configuring Polarity with Atmel-Abel™ 4.x and Abel™ 4.x

The optimization level best suited for Atmel PLDs is the default option – reduce by pin and auto polarity. This reduction level will take advantage of the polarity control when performing logic optimization one output at a time. This will override the ISTYPE 'NEG' and ISTYPE 'POS' used in Abel™ 3.x source files (check the user manual on backward compatibility for detail). Therefore, the 'NEG' and 'POS' extensions are not recommended.

The following examples have A, B, and C defined as inputs and OUT or !OUT as the output:

Case 1: (Combinatorial - no ISTYPE definition)

```
Declaration
OUT pin 14;
"assume 14 is an I/O pin
equations
OUT = A # B # C;
```

In this case, the compiler will consider both Figure 1 (on-set) and Figure 2 (off-set) and automatically select the implementation requiring fewer product terms for the same function. The outcome is represented by Figure 2. Since Figures 1 and 2 are each DeMorgan equivalent of the other, either one is logically correct.

Case 2: (Combinatorial - ISTYPE 'BUFFER')

```
Declaration
OUT pin 14 ISTYPE 'buffer';
"assume 14 is an I/O pin
equations
OUT = A # B # C;
```

In this case, the compiler will only consider the on-set because the ISTYPE 'BUFFER' overrides the automatic selection. The outcome is represented by Figure 1.

Case 3a: (Combinatorial - ISTYPE 'INVERT')

```
Declaration
OUT pin 14 ISTYPE 'invert';
"assume 14 is an I/O pin
equations
OUT = A # B # C;
```

In this case, the compiler will only consider Figure 2 (off-set) because the ISTYPE 'INVERT' overrides the automatic selection. The outcome is represented by Figure 2.

Case 3b: (Combinatorial - no ISTYPE definition)

```
Declaration
!OUT pin 14;
"assume 14 is an I/O pin
equations
OUT = A # B # C;
```

The compiler would pick Figure 3 to implement the logic because it takes fewer product terms. In Abel™ documentation, signals on the right side of the equation do not have "!" as part of their names. Abel™ preprocessor will remove the "!" from the pin name on the right side of the equation and replace all references on the left side with an additional "!". Logically, this does not change anything. It does, however, tend to create some confusion reading the .DOC files. In the source file, the user should still use whatever pin name is given in the declaration section. All references to the pin or .FB feedbacks will be adjusted by the software to reflect the changes automatically.

UV Erasable Programmable Logic Device

Application Note

5

Case 3c: (Combinatorial - ISTYPE 'INVERT')

```
Declaration
!OUT pin 14 ISTYPE 'invert';
"assume 14 is an I/O pin
equations
OUT = A # B # C;
```

The compiler would pick Figure 4 to implement the logic.

For combinatorial equations, it is best to leave out the ISTYPE statement and let the optimizer choose the best DeMorgan equivalent implementation.

Figure 1.

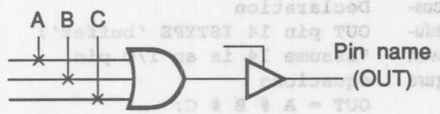


Figure 2.

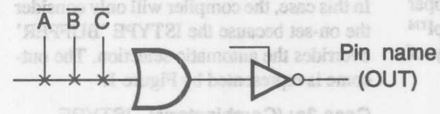


Figure 3.

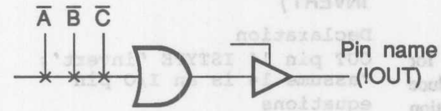
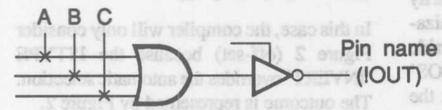


Figure 4.



Case 4: (Registered - no ISTYPE definition) Beware!

```
Declaration
OUT pin 14;
"assume 14 is an I/O pin
equations
OUT.d  = A # B # C;
OUT.c  = CLK;
OUT.ar = AR1;
```

The pre-processor will warn you for not specifying the ISTYPE of the output. In this case, the compiler will use the fewest product term implementation (Figure 6). This might not be what the user is expecting.(1)

Note:

1. Figure 5 and Figure 6 do not produce identical results. In Figure 5, at power up or after a reset, the output pin appears to be a "0." Unlike Figure 5, Figure 6 powers up and resets to a "1" on the output. Preset and preload behave differently between the two as well. In some applications where power-up state of a register is not important and it never resets or presets, Figures 5 and 6 become identical. Only in this case are they logically equivalent. When using a registered output, always specify the ISTYPE desired.

Case 5: (Registered - ISTYPE 'BUFFER')

```
Declaration
OUT pin 14 ISTYPE 'buffer';
"assume 14 is an I/O pin
equations
OUT.d  = A # B # C;
OUT.c  = CLK;
OUT.ar = AR1;
```

The compiler will only consider Figure 5 (on-set) because the ISTYPE 'BUFFER' overrides the automatic selection.

Case 6: (Registered - ISTYPE 'INVERT') Be careful!

```
Declaration
OUT pin 14 ISTYPE 'invert';
"assume 14 is an I/O pin
equations
OUT.d  = A # B # C;
OUT.c  = CLK;
OUT.ar = AR1;
```

The compiler will only consider Figure 6 (off-set) because the ISTYPE 'INVERT' overrides the automatic selection (see Note 1 on Case 4).

don't use.

```
Declaration
!OUT pin 14 ISTYPE 'buffer';
"assume 14 is an I/O pin
equations
OUT.d  = A # B # C;
OUT.c  = CLK;
OUT.ar = AR1;
```

The compiler will only consider Figure 7 (on-set) because the ISTYPE 'BUFFER' overrides the automatic selection.

```
Declaration
!OUT pin 14 ISTYPE 'invert';
"assume 14 is an I/O pin
equations
OUT.d  = A # B # C;
OUT.c  = CLK;
OUT.ar = AR1;
```

The compiler will only consider Figure 8 (off-set) because the ISTYPE 'INVERT' overrides the automatic selection. In Abel™ documentation, the pin name will be stripped of the "!". It will replace all pin name references with an additional "!" on the right-hand side of the equations.

Figure 5.

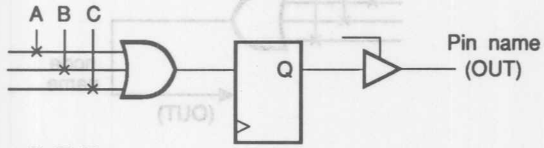


Figure 6.

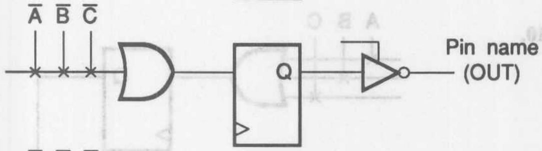


Figure 7.

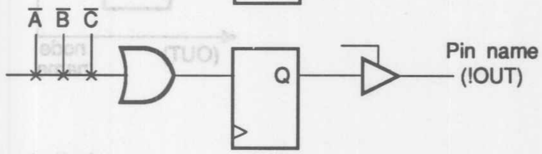
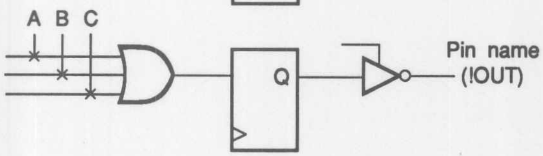


Figure 8.



Configuring Polarity with Internal Nodes

Internal nodes do not have programmable polarity control. Do not use any ISTYPE extensions. Think of it as "positive logic" only.

Case 1: (Figure 9)

```
Declaration
OUT node 50;
"assume 50 is an internal node
equations
OUT = A # B # C;
```

Case 2: (Figure 10)

```
Declaration
OUT node 50;
"assume 50 is an internal node
equations
OUT.d  = A # B # C;
OUT.c  = CLK;
OUT.ar = AR1;
```

The above example is the only legal method of assigning equations to nodes.

Figure 9.

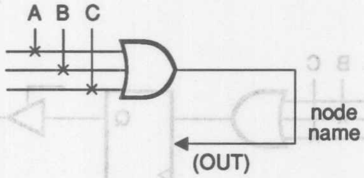
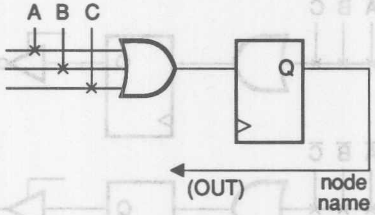


Figure 10.



Configuring Polarity with Cupl™

Note that Cupl™ has no "buffer/invert" ISTYPE statement. Output polarity is controlled by pin declaration versus equation polarity.

Case 1: (Combinatorial)

```
PIN 14 OUTC; /* assume 14 is an I/O pin */
OUTC = A # B # C;
```

The compiler would choose Figure 11. It does not choose the better DeMorgan equivalent automatically. If your equation does not fit, you should check to see if you can rewrite it as:

```
PIN 14 OUTC;
!OUTC = !A & !B & !C;
```

Case 2: (Combinatorial)

```
PIN 14 !OUTC; /* assume 14 is an I/O pin */
OUTC = A # B # C;
```

The compiler would choose Figure 12. The difference between the pin declaration (!OUTC) and the equation name (OUTC) tells the compiler to have an inverter on the output and to implement the equations as specified by the equation.

Case 3: (Registered)

```
PIN 14 OUTC; /* assume 14 is an I/O pin */
OUTC.d = A # B # C;
OUTC.ck = CLK;
```

The compiler would choose Figure 13. The pin name and the equation name are identical; the compiler does not place an inverter on the output.

Case 4: (Registered)

```
PIN 14 !OUTC; /* assume 14 is an I/O pin */
OUTC.d = A # B # C;
OUTC.ck = CLK;
```

The compiler would choose Figure 14. The difference between the pin declaration (!OUTC) and the equation name (OUTC) tells the compiler to have an inverter on the output and to implement the equations as specified by the equation. Cupl™ maintains the !OUTC on the pinout diagram documentation and equation name OUTC in the reduced equation portion of the documentation.

Both Abel™ and Cupl™ conventions for handling signal polarity have drawn praises and criticisms. Help on the software is readily available from Data I/O Corporation (Abel™), Logical Devices, Incorporated (Cupl™), and Atmel. Don't hesitate to call for help.

Figure 11.

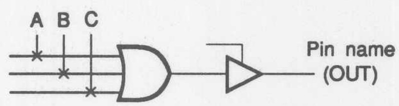


Figure 12.

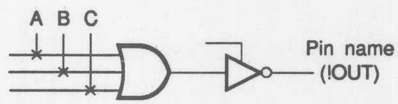


Figure 13.

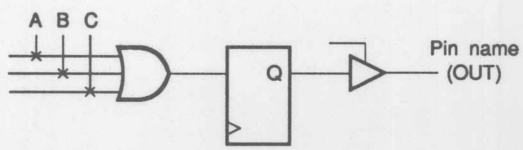
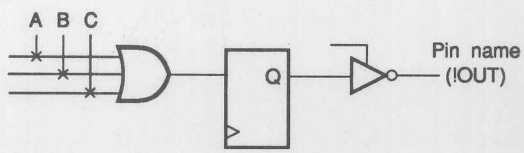


Figure 14.



Atmel PLDs' Architectures Simplify Timing Calculation

Programmable Logic Device

Application Note

Introduction

This application note shows different graphical timing models that can help the user visualize the A.C. timing of the various Atmel PLD families of devices. Because of their deterministic and path-independent delays, timing calculation becomes straight forward.

Atmel PLDs have regular AND-OR architecture which simplifies timing calculation. All the A.C. timing parameters are clearly stated in the data book. Even for complex designs it only takes a few minutes to calculate the delays by hand.

If the design engineer has access to tools such as the Atmel-ViewPLD, he/she can easily predict the performance of the PLD. PLD software packages with timing simulation capabilities let the design engineer know the performance of the PLD immediately after the design is entered and check the results of the timing simulator and quickly modify the design to meet the system timing requirements. Atmel offers a complete design entry package called Atmel-ViewPLD that has such a timing simulator.

Architectures/Timing Models

The AT22V10 represents the classic PAL™-type architecture with the programmable AND and fixed OR structure. A very small set of A.C. timing parameters can describe all the delays that occur in the implementation of register and combinatorial logic as shown in Figure 1.

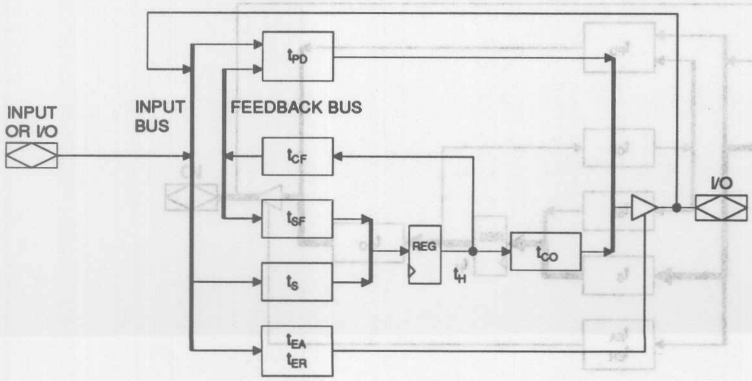
For example, an output is described in the following Boolean equation:

```
OUTPUT = A0 & !A1 & A2 # SELECT;
```

A0, A1, A2, and SELECT can be inputs or I/Os and the delay will be simply the time for the signal to propagate from a pin through the AND-OR array, the macrocell, and to the output pin. This is described by the A.C. parameter t_{PD}.

The ATV750 and ATV2500, with more advanced macrocells, maintain the same AND-OR structure as the AT22V10. Because of this, they can also be described by the same AT22V10 timing model. Even when using the buried registers found in the ATV750 and ATV2500, the method of calculation for delays stays the same. For ex-

Figure 1. AT22V10, ATV750, and ATV2500 Timing Model



ample, an Abel™ description of a binary counter may look like:

```
COUNT.d = COUNT.fb + 1;
```

The counter can be implemented using only the internal buried registers of the ATV750 or ATV2500. In this case the minimum cycle time will be equal to *t_{CF} (clock to feedback) + t_{SF} (feedback setup)*. See Figure 2. This is also equal to *1/(F_{MAX} internal)*.

Figure 3 shows the registered data path for a pin-to-pin delay, as might be described by output logic:

```
REG_A.d = A1 & B1 & !C1;
```

If A1, B1, and C1 are all signals from either input or I/O pins, then the minimum cycle time will be *t_S (setup time for input or I/O pin) + t_{CO} (clock to output)* which is equal to *1/(F_{MAX} external)*.

Figure 4 shows how data propagates for a typical Mealy state-machine, in which the state bits are inputs to combinatorial outputs:

```
COUNT.d = (COUNT.FB +1);
FULL = (COUNT.FB == ^HFF);
```

In this case, it will take FULL the delay of t_{CF} + t_{PD} to go from the rising edge of the clock driving the counter to the changing of FULL's output value.

Figure 5 is based upon Figure 1 with the addition of a few A.C. parameters to help describe the new features of the ATV5000 and ATV5100.

These devices have both synchronous and asynchronous modes of operation. With the addition of the synchronous clocking option, the devices perform at a higher clock rate. The A.C. parameters have either the suffix of "S" (synchronous) or "A" (asynchronous) to distinguish the two registered clocking options.

Input latch setup and hold time are additional requirements when the latch is used. If the latch is bypassed, no delay is added.

For these devices, the t_{PD} parameter is broken down further to show different delay paths separately. T_{PD1} and t_{PD2} are similar to the traditional t_{PD} parameter. T_{PD1} is the delay from any pin to any combinatorial output. T_{PD2} is the delay from internal feedback nodes to a combinatorial output pin.

Figure 2. ***1/(F_{MAX} internal)***

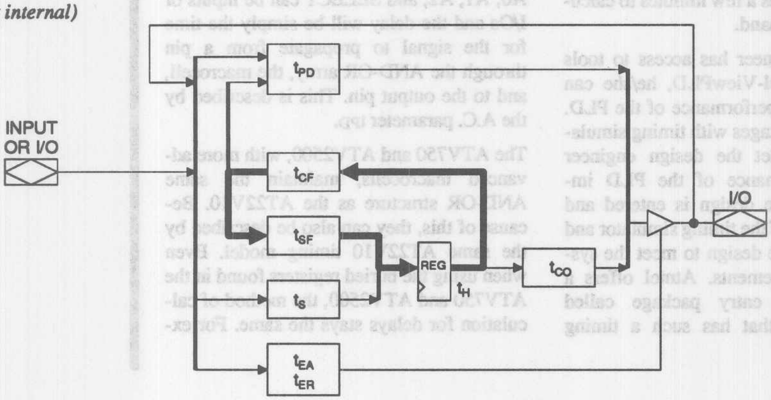
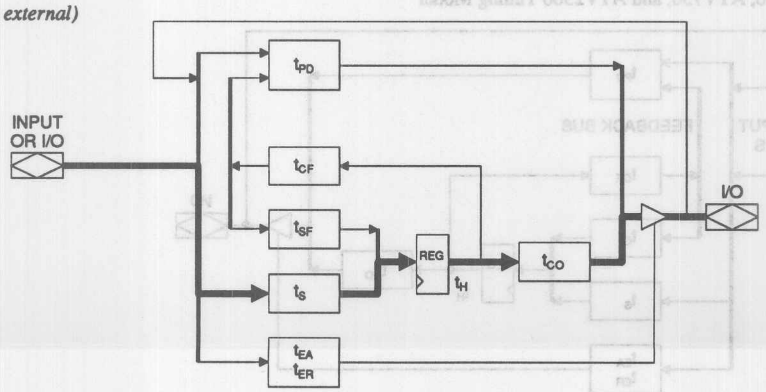


Figure 3. ***1/(F_{MAX} external)***



T_{PD3} is the delay from a pin to an internal combinatorial feedback.

T_{PD4} is the delay from an internal feedback, through the AND/OR array, to an internal combinatorial feedback.

The ATV5000 and ATV5100 Buried Logic Cells let the user configure internal combinatorial or registered feedbacks. Having a combinatorial feedback enables the designer to expand the logic without sacrificing any I/O pins. Also, the B sum term feedbacks in the ATV5000 and ATV5100 can take regional array inputs and make them available to other quadrants.

Figure 4. Mealy Machine Delay Path

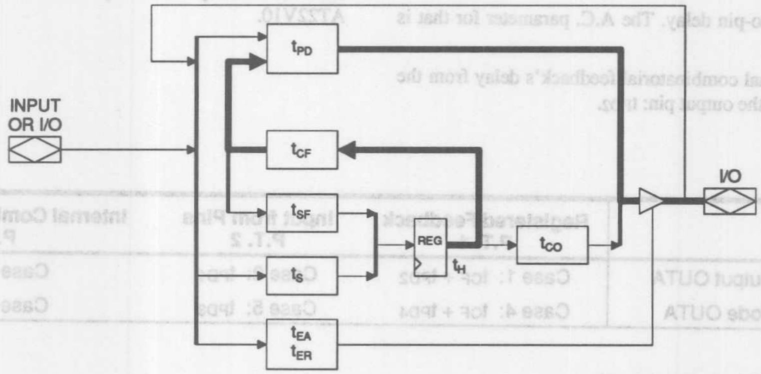
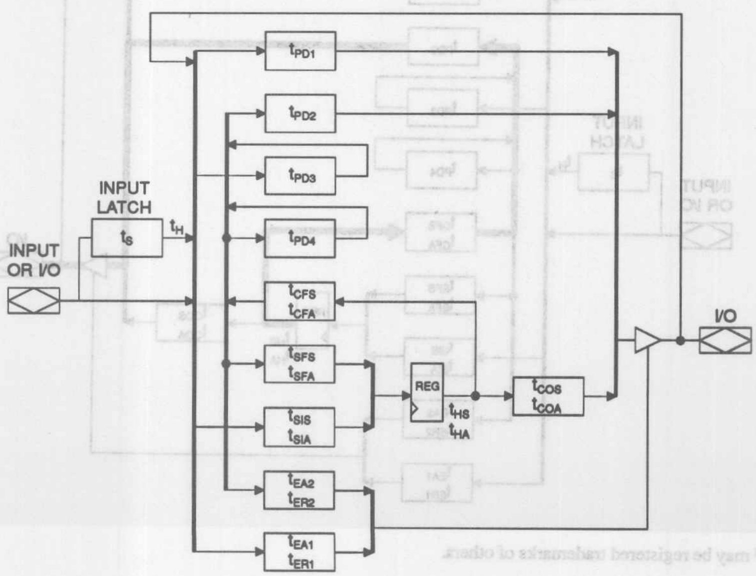


Figure 5. ATV5000 and ATV5100 Timing Model



The most straightforward way to determine the delays is to look at the documentation generated by PLD software after the design has been reduced and fitted. If a reduced equation looks like:

```
OUTA = WATCHDOG              "Product term 1
     # A1 & A2 & A3          "Product term 2
     # C_OUT;                "Product term 3
```

1. Determine whether the logic is registered or combinatorial.
2. Determine whether OUTA is an internal node or an output pin.

5

3. Find the source of each of the components that makes up the product terms.

To analyze the various cases, let's assume the following:

1. OUTA is combinatorial. We will look at OUTA implemented on an output pin versus OUTA implemented on a combinatorial node.
2. WATCHDOG is an internal registered node, A1 through A3 are directly from the inputs, and C_OUT is an internal combinatorial node (this covers all signal sources).

Table 1 summarizes the various timing requirements.

Case 1 (Figure 6) is the typical Mealy state machine where the internal state registers are decoded to form a combinatorial output. The total delay from clock to output is $t_{CF} + t_{PD2}$.

Case 2 is the pin-to-pin delay. The A.C. parameter for that is t_{PD1}.

Case 3 is an internal combinatorial feedback's delay from the AND/OR array to the output pin: t_{PD2}.

Case 4 is a "buried Mealy" where the internal state registers are decoded but not placed on an output pin. Instead the result is implemented on an internal combinatorial node where the logic is only useful internal to the design.

Case 5 is the delay from pin to internal combinatorial feedback: t_{PD3}.

Case 6 is the delay from one internal combinatorial node to another internal combinatorial node.

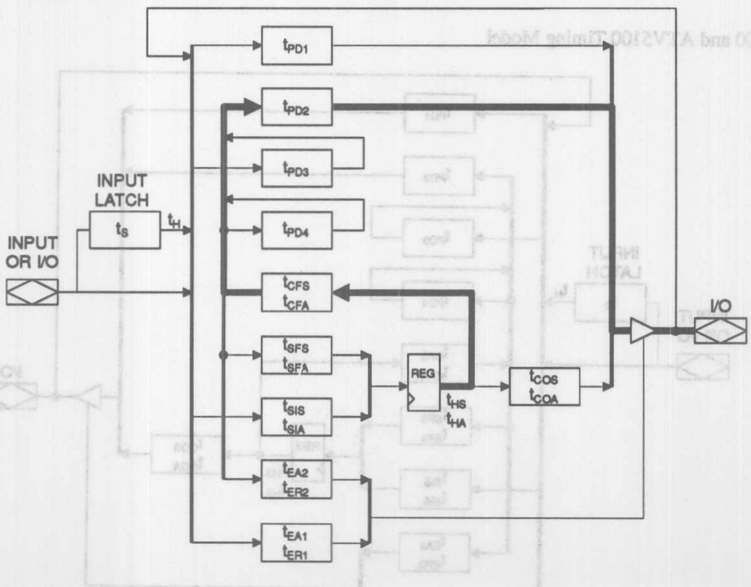
Conclusion

The graphical representation of the A.C. timing models illustrate how simple it is to determine the performance of logic implemented in a Atmel PLD. Atmel complex PLDs, even with their high pin counts and advanced features, have simple timing calculation. They aren't any harder to use than a common AT22V10.

Table 1.

	Registered Feedback P.T. 1	Input from Pins P.T. 2	Internal Combinatorial Node P.T. 3
Combinatorial Output OUTA	Case 1: $t_{CF} + t_{PD2}$	Case 2: t_{PD1}	Case 3: t_{PD2}
Combinatorial Node OUTA	Case 4: $t_{CF} + t_{PD4}$	Case 5: t_{PD3}	Case 6: t_{PD4}

Figure 6. Typical Mealy State Machine



PAL™ and Abel™ may be registered trademarks of others.

ATV2500 Application Example: Video Frame Grabber

Programmable Logic Device

Application Note

Introduction

This application note shows how the ATV2500 can be used to incorporate multiple control or logic functions into a single programmable logic device. The design example which is used is a simple NTSC video frame grabber. The ATV2500 is used to generate all of the control and addressing for the frame grabber. The application note includes a description of the frame grabber design and implementation using the ATV2500. The Abel™ source code for the ATV2500 is included for reference and is also available from the PLD applications group on floppy disk.

ATV2500 Description

The ATV2500 is a high density programmable logic device which features 24 I/O pins and 14 input-only pins. Each I/O pin is associated with a logic macrocell (see Figures 1 and 2). The output can be configured as either combinatorial or registered. Each macrocell contains two flip-flops, 12 product terms which can be split into three separate sum terms, and an output enable. Each flip-flop has a clock term and an asynchronous reset term. Groups of four or eight flip-flops each have a common synchronous preset product term.

Each macrocell has a feedback path from the pin and from each register. This makes it possible to bury both registers and use the pin for either a combinatorial output or an input pin. A global bus routes all pins and register feedbacks to every logic cell.

Frame Grabber Design Considerations

The basic idea behind a frame grabber is to sample and store a frame of video data. Once the data is stored, it can be re-displayed, enhanced or saved to a file. In this example, the input video signal is converted with an A/D

Figure 1. ATV2500 Output Logic, Registered [1]

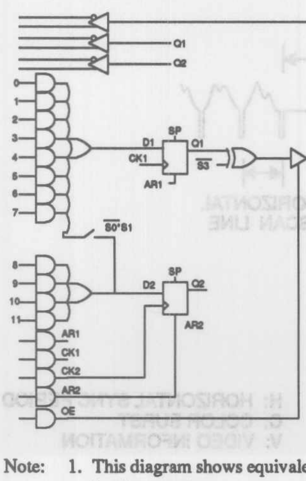
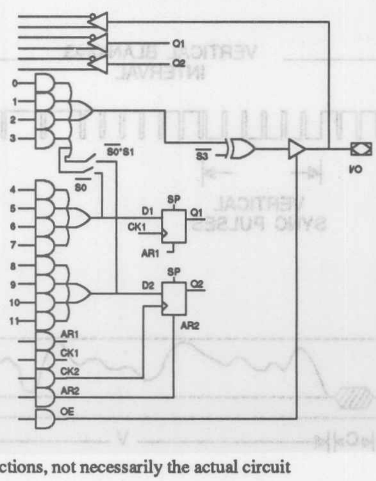


Figure 2. ATV2500 Output Logic, Combinatorial [1]



Note: 1. This diagram shows equivalent logic functions, not necessarily the actual circuit implementation.

5

video signal through a D/A converter.

Video Basics

The NTSC (National Television Standards Committee) composite video signal is the standard used by most television and video systems in North America. The signal is composed of four components: luminance (brightness), chrominance (color), audio and synchronization. An NTSC video image or *frame* is composed of 525 scan lines. Each frame is actually divided into two *fields* of 262.5 scan lines which are interlaced. The fields start with a vertical sync period followed by the scan lines (see Figure 3). Each scan line consists of a horizontal sync period, color burst and video information (see Figure 4).

Timing

The frame grabber samples the entire video frame, including the horizontal and vertical sync periods. Then, in order to generate a video image from the stored data, the data is simply converted back to an analog signal at the same rate it was sampled.

The refresh rate for each field is 59.94 Hz (16.683 ms per field), which means the refresh rate for the whole frame is 29.97 Hz (33.366 ms per frame). The number of bits in each sample and the sampling frequency determine the resolution of the reconstructed video image. An 8-bit sample size was chosen for this example since 8-bit A/D converters are readily available. An 8-bit sample size will allow 256 levels of intensity, which is plenty for the purposes of this design. In order to generate a reasonable image, the sample frequency should be at least twice the NTSC color burst frequency of 3.579545 MHz. A sample frequency of 7.5 MHz was chosen, which is a little more than twice the color burst frequency.

The total number of samples required for each frame will be 253245 (*7.5 MHz x 16.683 ms x 2*).

is depressed, the frame grabber will pass the video signal through to the output. When the button is released, the data is sampled. The converted video signal is monitored to detect a vertical sync. Once the vertical sync is detected, 253245 samples from the video input are stored. Following the sampling, the addresses are continuously cycled creating a frozen image. The captured frame will be displayed until the button is depressed again, causing another frame of data to be sampled and stored.

It is not necessary to start sampling during the vertical sync period since 253245 samples will contain an entire frame. As the addresses are cycled, the vertical sync portion of the video signal will be generated every 16.683 ms. However, if the data were used for any purpose other than just display, it would be more convenient to have the data start at a known point in the video signal.

Frame Grabber Implementation

The schematic for the frame grabber is shown in Figure 5. The three basic functions are: D/A and A/D conversion, control and address generation, and storage for the sampled data.

For the A/D and D/A conversions, the Samsung KSV3100A was selected. It has both an 8-bit A/D converter and 10-bit D/A converter, along with the necessary pre-amplifier and input clamping circuit in a single device. The KSV3100A device is connected as in the recommended operating circuit in the data sheet. The 7.5 MHz system clock is used as the clock for both the A/D and D/A conversions.

The ATV2500 is used to implement all of the control and addresses for the frame grabber. The functions include a vertical sync detector, a control state machine and a RAM address counter. It receives the mode signal from the switch, the 8-bit data bus from the A/D converter and the 7.5 MHz system clock

Figure 3. Field Timing

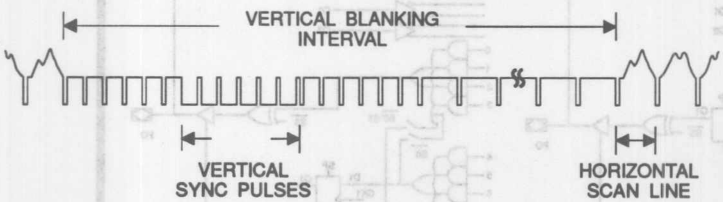
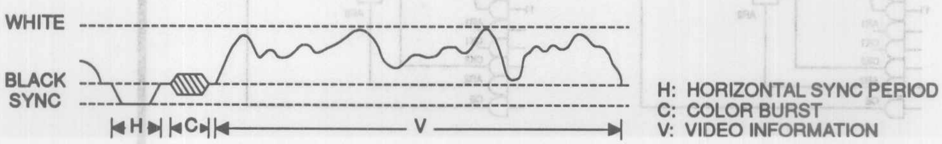


Figure 4. Horizontal Scan Line



signal. It generates the bus control signals and 18-bit address for the RAM.

A 256K x 8 static RAM module is used for the frame buffer memory. A write enable signal from the controller allows the converted data to be stored in the RAM during sampling. An output enable signal is used to enable/disable the RAM onto the data bus. External buffers are used to enable the data from the output of the A/D back into the D/A when the video signal is passed through or the data is being sampled.

ATV2500 Control and Address Functions

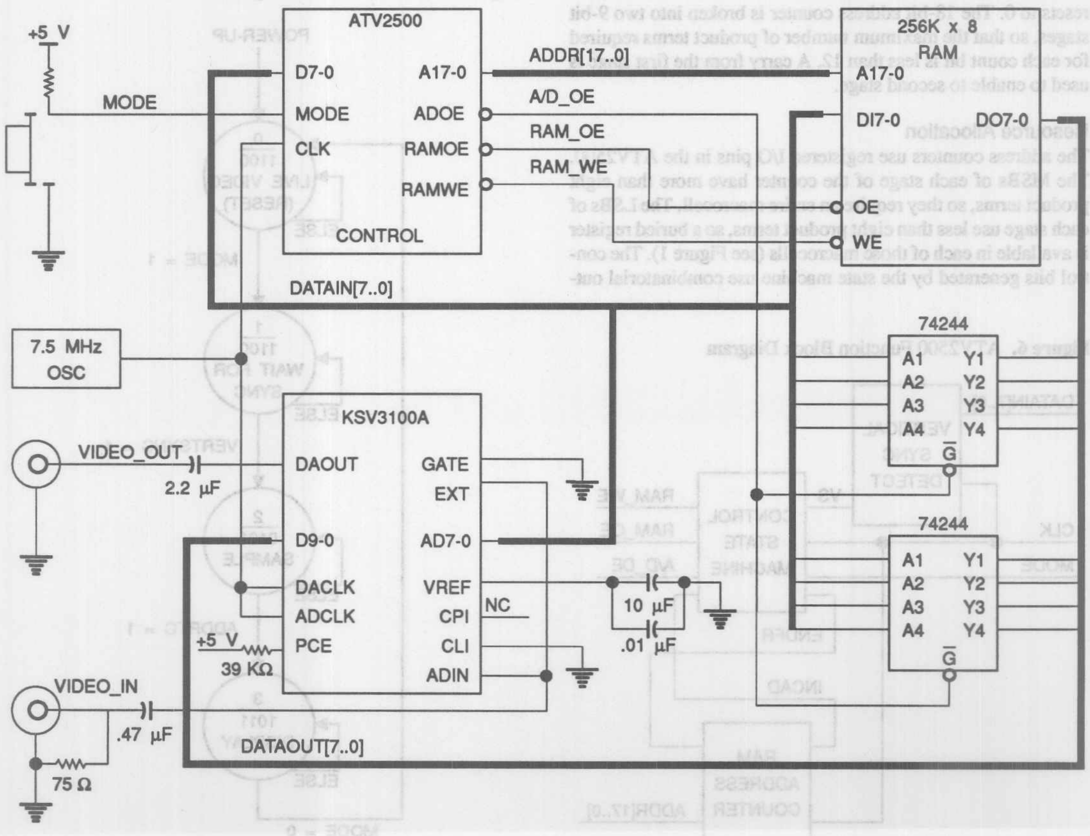
The ATV2500 is used to implement the control and address functions. These include a vertical sync detector, a control state machine and the RAM address counter. Figure 6 shows a block diagram of the ATV2500 functions.

Vertical Sync Detector

During the horizontal and vertical sync periods, the signal drops to a sync level which is lower than any other portion of the signal. This level will be clamped in the A/D converter and will become the zero value when it is converted. The horizontal sync pulse is about 4.7 µs, which will correspond to about 35 samples. The entire vertical sync period is 20 times the horizontal scan width. Within that period, there are pulses of around 31 ms when the signal is at the sync level. Each of these pulses will correspond to about 238 samples.

The vertical sync detector is a 7-bit counter which will count every sample with a zero value. If it counts 128 such samples, then it has detected one of the pulses in the vertical sync period. The counter is divided into two parts, so that the maximum number of product terms required for any count bit is four. A carry bit from the first stage is used to enable the second stage.

Figure 5. Frame Grabber Schematic



The counter will increment whenever the input data is zero. A non-zero value will cause the counter to reset. When the count reaches 127, the VS signal is asserted.

State Machine

A state machine generates internal control for the address generation and external control signals for the RAM. The state diagram is shown in Figure 7. When the MODE input changes, indicating that the frame grabber should sample a new frame, the state machine waits for the VS signal. After the vertical sync is detected, it sets up the control signals to write the sampled data for the entire frame. When the address counter reaches the correct number of samples, the state machine changes the control signals to read the sampled data. The addresses are cycled so that the sampled frame is continuously displayed.

Address Counter

The address counter is controlled by the INCAD control signal from the state machine. If the INCAD signal is asserted, the address will increment until the counter reaches the final address for the correct number of samples. When the INCAD signal is not asserted, or the final address is reached, the counter resets to 0. The 18-bit address counter is broken into two 9-bit stages, so that the maximum number of product terms required for each count bit is less than 12. A carry from the first stage is used to enable to second stage.

Resource Allocation

The address counters use registered I/O pins in the ATV2500. The MSBs of each stage of the counter have more than eight product terms, so they require an entire macrocell. The LSBs of each stage use less than eight product terms, so a buried register is available in each of those macrocells (see Figure 1). The control bits generated by the state machine use combinatorial outputs with less than four product terms, so two buried registers are available in each of those macrocells (see Figure 2). The state bits and vertical sync detect counter bits are assigned to available buried registers.

Assignment of the I/O pins is only dependent on the board layout, since there are no signal routing limitations in the ATV2500. Buried logic can be assigned to any available resources. Table 1 shows a worksheet used to allocate the logic functions in this application to the ATV2500 resources. The shaded boxes indicate resources which are used by another signal generated in the same macrocell. The empty boxes indicate available resources. In this design, four input pins, one I/O pin and associated macrocell plus 13 additional buried registers are available for expansion or design changes.

Notes on Test Vectors

In order to test that the address counter rolls over at the correct count, more than 253245 test vectors would be required, which

Figure 6. ATV2500 Function Block Diagram

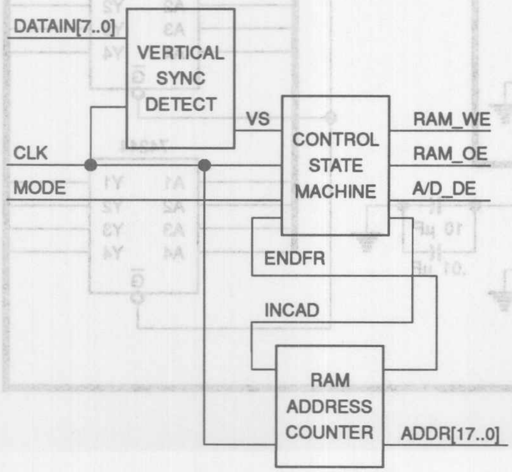
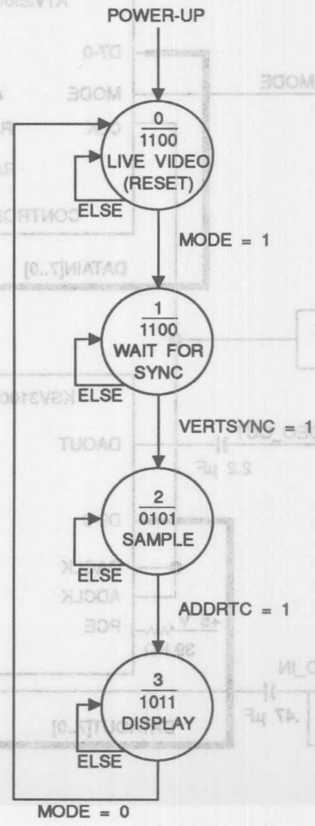


Figure 7. Controller State Diagram



Note: Outputs: RAMWE, RAMOE, ADOE, INCAD

far exceeds the number of vectors allowed in Abel™. There are a few ways to get around this problem.

The first method is to use the register preload function in PLASIM™. The counter can be preloaded with a count near to the maximum count, and then allowed to roll over. However, since JEDSIM™ does not support the preload function the JEDEC file is not checked.

Another method is to change the terminal count value to a smaller value, so that the counter would reach the terminal count in fewer vectors. Since the equations are altered, this method could only be used for Abel™ simulation.

A third method is to use the register synchronous presets to load the counter. This is the method that was used in this design. The registers which use the preset are allocated to the same ATV2500 synchronous preset groups. A spare input pin (PRELD) is used to control the preset in the test vectors. Using this method, the vectors can also be used on the programmer.

Summary

In this example, the address and control functions are greatly simplified by using a single complex programmable logic device. The ATV2500 is ideal for this application since it has enough inputs to accept the control and data signals, and the necessary I/Os to generate the RAM addresses and control. The ability to combine or separate the sum terms in each macrocell allows for maximum usage of the available resources. The functions which require large numbers of product terms do not have to be split into smaller pieces. The functions which require fewer numbers of product terms do not waste an entire macrocell. The leftover buried logic in those macrocells can be used for the control state machine and the vertical sync detector.

Table 1. ATV2500 Pin/Node Assignment Worksheet Example

Input	Signal	Input	Signal
1	CLK	21	D4
2	MODE	22	D5
3		23	D6
17	DO	37	D7
18	D1	38	
19	D2	39	
20	D3	40	(PRELD)

I/O	Signal	Q2	Signal	Q1	Signal
4	A0	41	SYNC0	217	
5	A1	42	SYNC1	218	
6	A2	43	SYNC2	219	
7	A3	44	SYNC3	220	
8	A4	45	SYNCARRY	221	
9	A5	46	SYNC4	222	
11	A8	47		223	
12	A10	48		224	
13	A6	49	SYNC5	225	
14	A7	50	SYNC6	226	
15	A9	51	ENDFR	227	
16	A13	52	ADCARRY	228	
24	A11	53		229	
25	A12	54		230	
26	A14	55		231	
27	A15	56		232	
28	A16	57		233	
29	A17	58		234	
31	VS	59	ST0	235	
32	RAMWE	60	ST1	236	
33	RAMOE	61		237	
34	ADOE	62		238	
35	INCAD	63		239	
36		64		240	

Abel™ Source Code

```
module CONTROL ;
title 'Application example for the ATV2500 - Frame Grabber Controller
Atmel Corporation PLD - (408)436-4333 PLD Applications Hotline
Wendey Mueller - January 10, 1992'

CONTROL device 'P2500';

CLK,MODE             pin 1,2; "system clock and mode signal from button
D0,D1,D2,D3          pin 17,18,19,20; "digitized video signal from A/D
D4,D5,D6,D7          pin 21,22,23,37;

A0,A1,A2,A3,A4       pin 4,5,6,7,8 istype 'reg,buffer'; "RAM address outputs
A5,A6,A7,A8,A9       pin 9,13,14,11,15 istype 'reg,buffer';
A10,A11,A12,A13,A14  pin 12,24,25,16,26 istype 'reg,buffer';
A15,A16,A17          pin 27,28,29 istype 'reg,buffer';
RAMWE,RAMOE          pin 32,33 istype 'com,buffer'; "RAM control signals
ADOE,INCAD           pin 34,35 istype 'com,buffer'; "address counter control
VS                   pin 31 istype 'com,buffer';    "vertical sync detected

ADCARRY,SYNCARRY     node 52,45 istype 'reg,buffer'; "counter carry signals
ENDFR                node 51 istype 'reg,buffer';    "end of frame detected
ST0,ST1              node 59,60 istype 'reg,buffer'; "state bits
SYNC0,SYNC1,SYNC2    node 41,42,43 istype 'reg,buffer';
SYNC3,SYNC4,SYNC5    node 44,46,49 istype 'reg,buffer'; "vertical sync detect
SYNC6                node 50 istype 'reg,buffer';     "counter

"Use spare input pin as preset signal for testing counter

PRELD                pin 3;

C,X,Z,H,L,P = .C.,.X.,.Z.,1,0,.P.;

"Create buses

DATA   = [D7..D0];        "data bus
ADDR   = [A17..A0];       "address counter
ADDRA  = [A8..A0];        "address counter LSB
ADDRB  = [A17..A9];       "address counter MSB
SYNC   = [SYNC6..SYNC0];  "vertical sync detect counter
SYNCA  = [SYNC3..SYNC0];  "vertical sync detect counter LSB
SYNCB  = [SYNC6..SYNC4];  "vertical sync detect counter MSB
STMACH = [ST1,ST0];       "state bits

"Define states

S0  = [0,0];
S1  = [0,1];
S2  = [1,0];
S3  = [1,1];

EQUATIONS

"256K address counter - increment when INCAD is true and address
"                       has not reached end of frame, otherwise reset

ADDRA.D = ((ADDRA.FB + 1) & INCAD & !ENDFR); "LSB stage
ADDRA.CK = CLK;

ADCARRY.D = (ADDRA.FB == 510); "synchronous carry bit from 1st stage
ADCARRY.CK = CLK;
```

```
ADDRB.D = ((ADDRB.FB + 1) & INCAD & !ENDFR & ADCARRY) "MSB stage
        # (ADDRB.FB & INCAD & !ENDFR & !ADCARRY);
ADDRB.CK = CLK;

ENDFR.D = (ADDR.FB == 253243); "set ENDFR when address reaches 253243
ENDFR.CK = CLK;

"Use synchronous preset to simplify testing the address counter by
"preloading it with a value close to the last address.

A17.SP = PRELD;
A16.SP = PRELD;
A15.SP = PRELD;
A14.SP = PRELD;
A12.SP = PRELD;
A11.SP = PRELD;
A10.SP = PRELD;
A8.SP  = PRELD;

"Vertical sync detect counter - increment if data is 0, otherwise reset

SYNCA.D = (SYNCA.FB + 1) & (DATA == 0); "LSB stage
SYNCA.CK = CLK;

SYNCARRY.D = (SYNCA.FB == 14); "synchronous carry bit from 1st stage
SYNCARRY.CK = CLK;

SYNCB.D = (SYNCB.FB + 1) & (DATA == 0) & SYNCARRY  "MSB stage
        # SYNCB.FB & (DATA == 0) & !SYNCARRY;
SYNCB.CK = CLK;

VS = (SYNC.FB == 127); "set vertical sync detected bit if count reaches 128

STMACH.CK = CLK;

"State Machine Controller -
"
"Inputs: MODE,VS,ENDFR
"Outputs: RAMWE,RAMOE,ADOE,INCAD

STATE_DIAGRAM STMACH

STATE S0:                        "Reset, live video
   RAMWE = 1;
   RAMOE = 1;
   ADOE = 0;
   INCAD = 0;
   IF (MODE) THEN S1             "If capture mode, go to state 1
   ELSE S0                       "else wait for mode change

STATE S1:                        "Wait for vertical sync signal
   RAMWE = 1;
   RAMOE = 1;
   ADOE = 0;
   INCAD = 0;
   IF (VS) THEN S2               "Vertical sync detected, start sampling data
   ELSE S1                       "else wait for vertical sync

STATE S2:                        "Sample video data
   RAMWE = 0;
   RAMOE = 1;
   ADOE = 0;
   INCAD = 1;
```

```
   IF (ENDFR) THEN S3          "Frame capture complete
   ELSE S2                     "else continue sampling

STATE S3:                      "Display frame data
   RAMWE = 1;
   RAMOE = 0;
   ADOE = 1;
   INCAD = 1;
   IF (!MODE) THEN S0          "Reset, display live video
   ELSE S3                     "else continue to display frame data

@RADIX 16;
@CONST ACNT = 1;
@CONST SCNT = 0;

TEST_VECTORS (
[CLK,MODE,DATA,PRELD] - [SYNC,VS,STMACH,RAMWE,RAMOE,ADOE,INCAD,ADDR, ENDFR])

"check that the vertical sync detector resets if data is not 0 and set mode
"to start sample and display sequence
[ 0,  0,  0,   0   ] - [00,  0, 0,    1,    1,    0,   0,     00000,0    ];
[ C,  0,  0,   0   ] - [01,  0, 0,    1,    1,    0,   0,     00000,0    ];
[ C,  0,  1,   0   ] - [00,  0, 0,    1,    1,    0,   0,     00000,0    ];
[ C,  1,  1,   0   ] - [00,  0, 1,    1,    1,    0,   0,     00000,0    ];

"simulate vertical sync
@REPEAT 7E {
@CONST SCNT = SCNT + 1;
[ C,  1,  0,   0   ] - [SCNT,0, 1,    1,    1,    0,   0,     00000,0    ]; }
[ C,  1,  0,   0   ] - [7F,  1, 1,    1,    1,    0,   0,     00000,0    ];
[ C,  1,  0,   0   ] - [00,  0, 2,    0,    1,    0,   1,     00000,0    ];
[ C,  1,  0,   0   ] - [01,  0, 2,    0,    1,    0,   1,     00001,0    ];

"state machine has detected vertical sync and started sampling.
@REPEAT 0FE {
@CONST ACNT = ACNT + 1;
[ C,  1,  1,   0   ] - [00,  0, 2,    0,    1,    0,   1,     ACNT, 0    ];}
[ C,  1,  1,   0   ] - [00,  0, 2,    0,    1,    0,   1,     00100,0    ];
[ C,  1,  1,   0   ] - [00,  0, 2,    0,    1,    0,   1,     00101,0    ];

"Use preset to preload a count value near the largest address value.
[ C,  1,  1,   1   ] - [00,  0, 2,    0,    1,    0,   1,     3DD02,0    ];
[ C,  1,  1,   0   ] - [00,  0, 2,    0,    1,    0,   1,     3DD03,0    ];

"Allow counter to reach the largest address and roll over. State machine
"then changes controls to display sampled data.
@CONST ACNT = 3DD03;
@REPEAT 037 {
@CONST ACNT = ACNT + 1;
[ C,  1,  1,   0   ] - [00,  0, 2,    0,    1,    0,   1,     ACNT, 0    ];}
[ C,  1,  1,   0   ] - [00,  0, 2,    0,    1,    0,   1,     3DD3B,0    ];
[ C,  1,  1,   0   ] - [00,  0, 2,    0,    1,    0,   1,     3DD3C,1    ];
[ C,  1,  1,   0   ] - [00,  0, 3,    1,    0,    1,   1,     00000,0    ];
[ C,  1,  1,   0   ] - [00,  0, 3,    1,    0,    1,   1,     00001,0    ];
[ C,  1,  1,   0   ] - [00,  0, 3,    1,    0,    1,   1,     00002,0    ];

END ;
```

ATV5000 Application Example: DMA Controller

Programmable Logic Device

Application Note

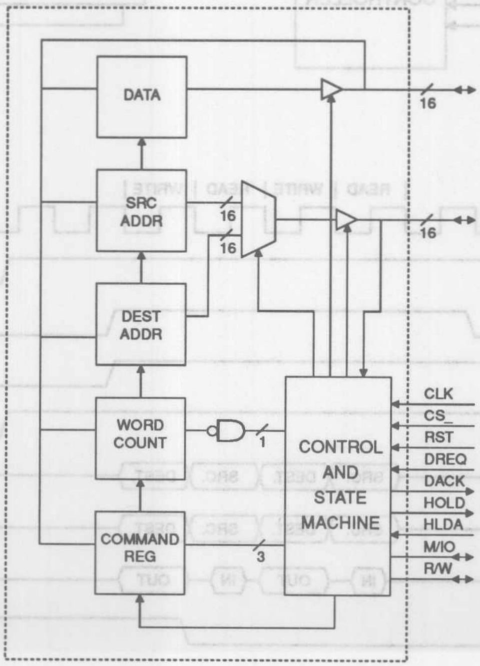
Introduction

This application note shows how the ATV5000 complex programmable logic device can be used to implement a simple DMA Controller. It includes a description of the DMA controller function and the implementation using an ATV5000. The Abel™ source code is included for reference, and is also available from the factory on floppy disk.

A Direct Memory Access (DMA) controller is a peripheral device used in a CPU system to perform block data transfers between memories and I/O devices. It generates the addresses and control necessary to perform the transfer. The DMA controller improves system performance by transferring data directly between devices instead of using the CPU.

The ATV5000 is a high-density programmable logic device, featuring 52 I/O logic cells and 128 flip-flops. This application takes advantage of the buried registers to store and increment or decrement the addresses and word count. The counters are easily implemented with minimal logic by using T-type registers. The high I/O count can easily accommodate the 16-bit bi-directional data and address buses.

Figure 1. Internal Block Diagram



5

DMA Controller Description

The DMA controller is used to transfer blocks of data between system memory and other memory or I/O devices. Figure 1 shows the basic block diagram for the DMA controller. The interface to the CPU consists of an address bus, a data bus, and some control signals. These signals are used by the CPU to load initialization data into the DMA controller registers. When the DMA controller is granted control of the buses, the same signals are used to transfer the data. This example uses i386™ type bus control signals. Figure 2 shows how this DMA controller would be used in a CPU system.

The transfer is initiated when the CPU loads the starting source and destination addresses, word count, and control word into the DMA controller registers. The external device asserts the DMA request signal to request a DMA. The DMA controller then requests control of the system buses and control signals. When the CPU grants the bus request, the DMA controller acknowledges the DMA request and starts the DMA.

The data is read in on the data bus and then latched and written back out. The source address or destination address is multiplexed onto the address bus during the read cycle or write cycle. The system bus control signals are used by the DMA controller as control signals for the memory or I/O device. The DMA controller transfers the requested number of words and then relinquishes bus control. The DMA cycle timing is shown in Figure 3.

Figure 2. System Block Diagram

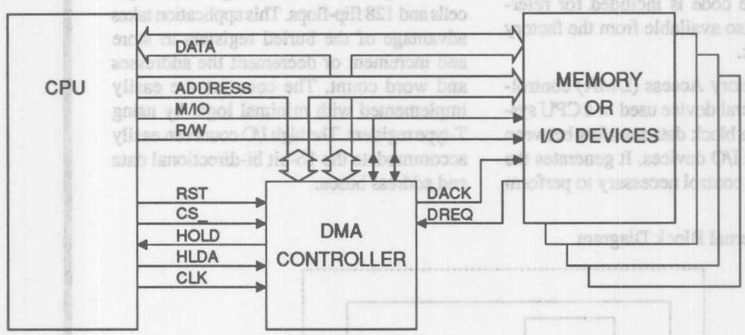
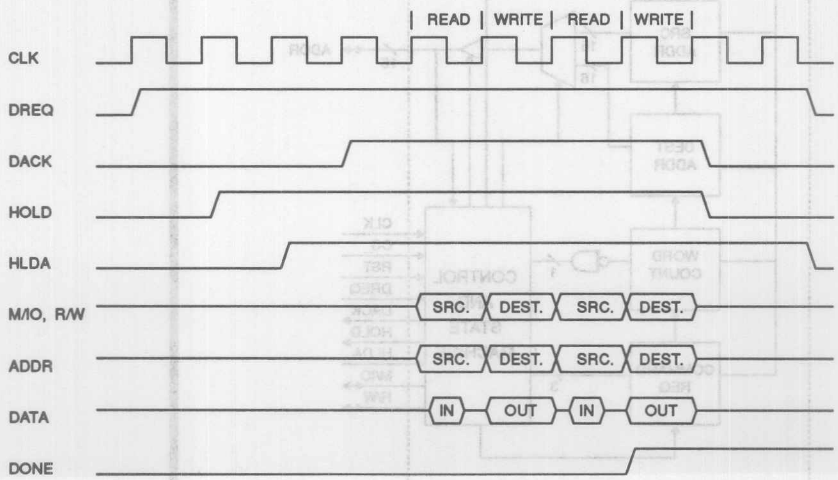


Figure 3. DMA Timing



ATV5000 Description

The ATV5000 is 68-pin high-density programmable logic device, which features 52 I/O pins and eight input-only pins. Each I/O pin is associated with a logic macrocell (see Figures 4, 5, and 6). Each macrocell has an input latch, two flip-flops, 13 product terms which can be split into three separate sum terms, and an output enable term. The I/O pin can be driven with either a combinatorial or registered output. Each flip-flop has a clock term, asynchronous reset term, and asynchronous preset term, and can be configured as either a D-type or T-type flip-flop.

In addition, there are 24 buried logic cells (see Figure 7). Each buried logic cell can be configured as registered or combinatorial. If registered, each flip-flop has a clock term, asynchronous reset term, and synchronous preset term, and can be configured as a D-type or T-type flip-flop.

The ATV5000 is divided into four quadrants with 13 I/O macrocells and six buried logic cells each (see Figure 8). A universal bus routes signals to all four quadrants, and a regional bus routes signals within each quadrant. The regional buses contain the true and false feedback signals from each register, from the buried logic cells, and from the eight input-only pins. The universal bus contains the regional bus inputs plus the true and false signals from each I/O pin.

Figure 4. Logic Cell, Two Buried Registers, Combinatorial to I/O Cell

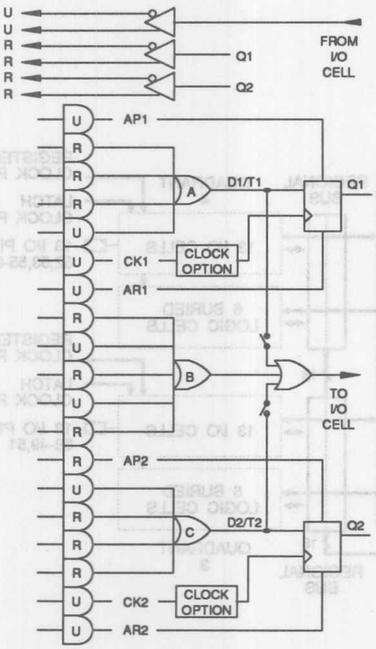


Figure 5. Logic Cell with Combinatorial Sum Terms, Register to I/O Cell

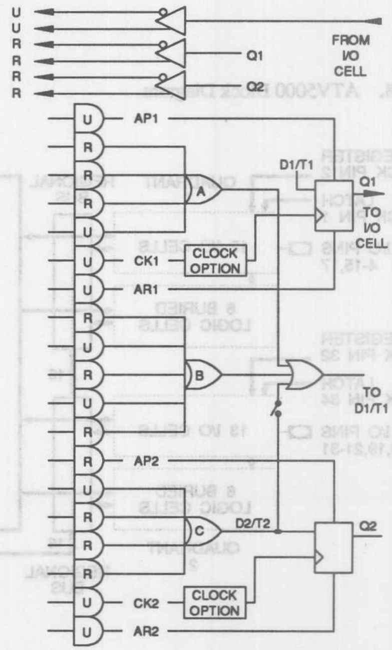


Figure 6. I/O Pin Logic

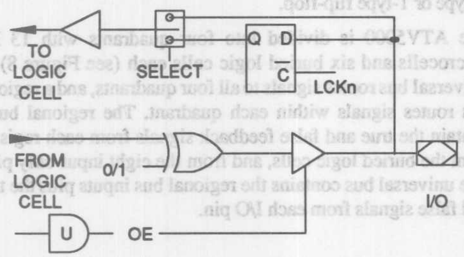


Figure 7. Buried Logic Cells

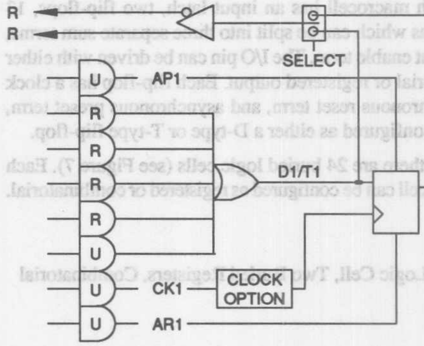
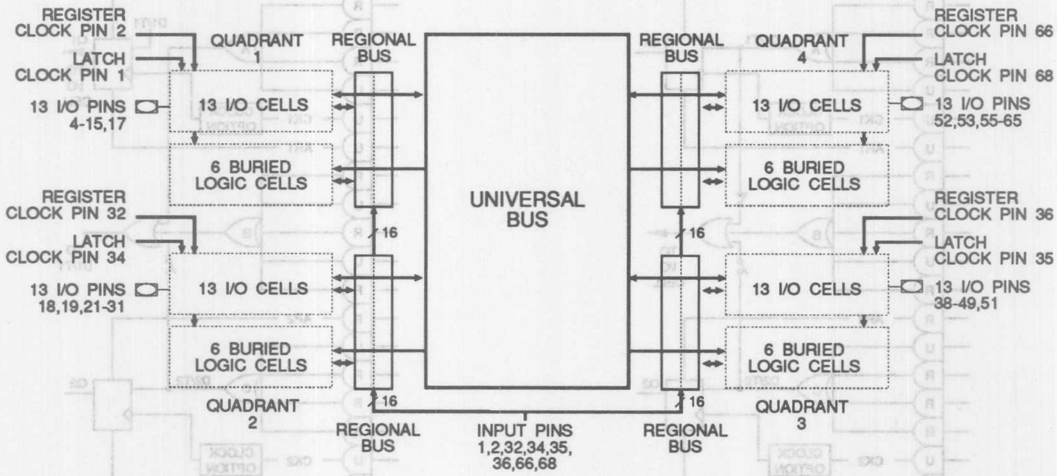


Figure 8. ATV5000 Block Diagram



DMA Controller Implementation

The DMA controller consists of three basic functions: system interface, internal address and word counters, and a state machine controller.

System Interface

The system interface consists of a 16-bit bi-directional data bus, a 16-bit bi-directional address bus, bi-directional bus control signals, and handshaking signals to request bus control.

The data and address buses are assigned to I/O pins in the ATV5000. When the CPU controls the system buses, the buses are used as inputs to receive initialization data. When the DMA controller controls the system buses, the buses are bi-directional. The data bus is tri-stated while the source data is read and registered, and then enabled to write the data to the destination. The address bus outputs are enabled and multiplex the source or destination address onto the bus.

The CLK and RST signals are the CPU system clock and reset signals. The CLK signal is used as the clock for all registers, so that all data transfers are synchronized to the system clock. The CS_ selects whether the DMA controller is being addressed for initialization. Both signals are assigned to input pins in the ATV5000, so that the signals are available on the regional buses in all quadrants.

The M/IO and R/W signals are used to control the bus activity. For initialization, the R/W signal enables a write and the LSB bits of the address bus are decoded to select which register is to be accessed. During a DMA transfer, the M/IO selects the source or destination as an I/O device or a memory and the R/W signal selects either a read or write operation. Both signals are assigned to I/O pins in the ATV5000, since they are used as inputs during initialization and outputs during a DMA cycle.

There are two sets of handshaking signals: the DMA request and acknowledge (DREQ and DACK), and the bus request and acknowledge (HOLD and HLDA). The DREQ and HLDA are assigned to input pins in the ATV5000. The HOLD and DACK signals are assigned to I/O pins in the ATV5000, since they are signals output by the controller.

Address and Word Counters

The address and word counters store the starting addresses, transfer word count, and command word. All use buried registers in the ATV5000, since the data does not need to be directly output. The registers are used as counters which load and then increment or decrement. The load and count functions are controlled by the bus control signals and the state machine.

The registers are configured as T-type flip-flops, so that a minimum number of product terms are required to implement the counter functions. Loading is accomplished by XORing the output of the register with the load data, causing the T flip-flop to toggle if the data does not match.

The outputs of the address registers are multiplexed onto the address bus to provide source and destination addresses for the DMA transfer. After each transfer, the addresses are incremented and the word count is decremented. When the word count reaches zero, the DONE signal is sent to the state machine, indicating that the transfer is complete.

The command word is decoded and used by the state machine to determine what type of DMA will be performed. The data is only used by the state machine, so it can be stored in buried registers in the ATV5000. The word consists of three bits: two bits which indicate whether the source and destination are memory or I/O devices, and a third bit which enables the DMA to start. After the DMA is complete, the enable bit is reset.

State Machine

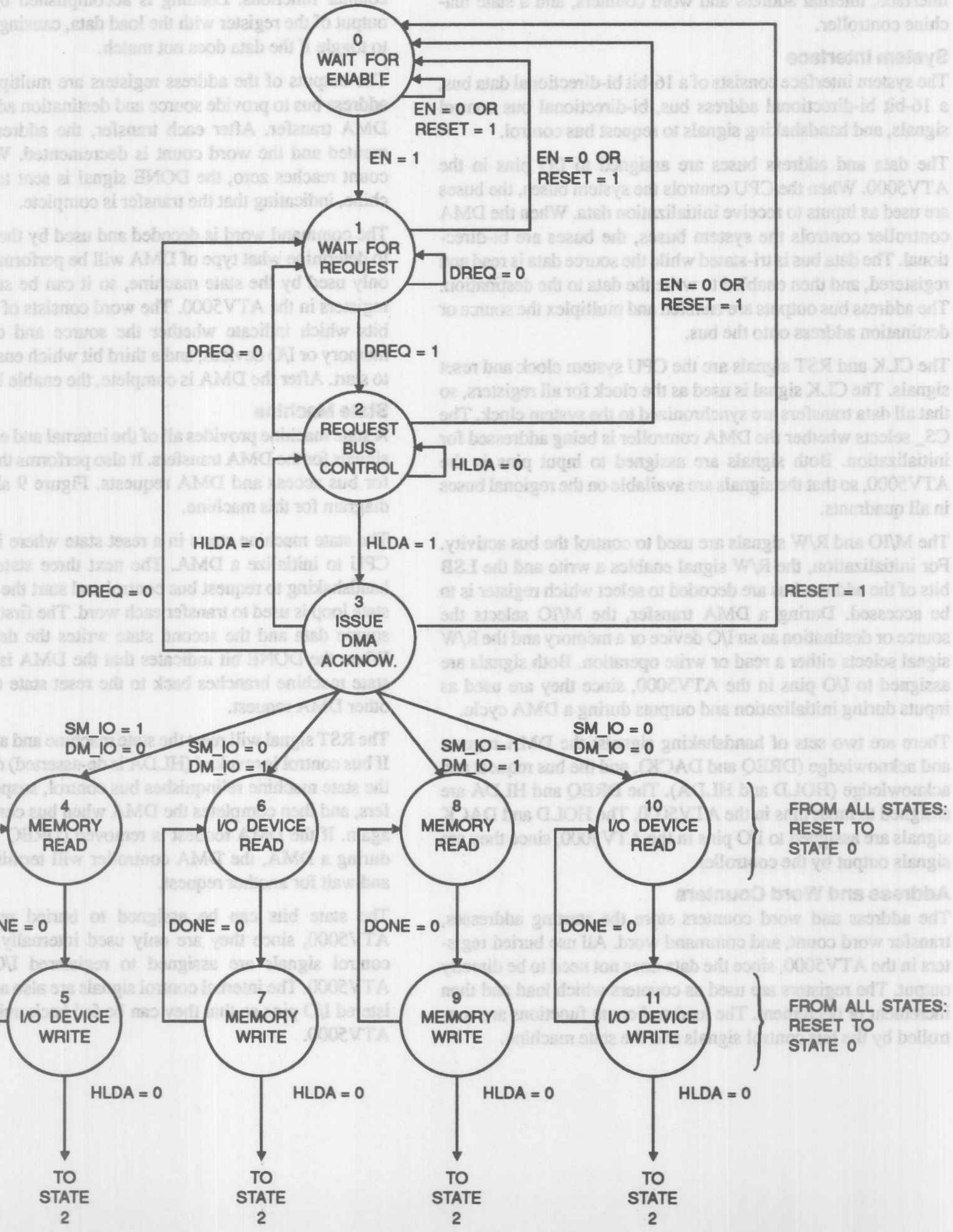
A state machine provides all of the internal and external control signals for the DMA transfers. It also performs the handshaking for bus access and DMA requests. Figure 9 shows the state diagram for this machine.

The state machine starts in a reset state where it waits for the CPU to initialize a DMA. The next three states perform the handshaking to request bus control and start the DMA. A two-state loop is used to transfer each word. The first state reads the source data and the second state writes the destination data. When the DONE bit indicates that the DMA is complete, the state machine branches back to the reset state to wait for another DMA request.

The RST signal will reset the state machine and abort the DMA. If bus control is revoked (HLDA is de-asserted) during a DMA, the state machine relinquishes bus control, suspends the transfers, and then completes the DMA when bus control is granted again. If the DMA request is removed (DREQ is de-asserted) during a DMA, the DMA controller will terminate the DMA and wait for another request.

The state bits can be assigned to buried registers in the ATV5000, since they are only used internally. The external control signals are assigned to registered I/O pins in the ATV5000. The internal control signals are also assigned to registered I/O pins so that they can be fed back universally in the ATV5000.

Figure 9. State Diagram



ATV5000 Resource Allocation

The goal in allocating functions to the available resources in the ATV5000 is to group functions which interface each other into the same quadrant. This will minimize the number of signals which must be routed on the universal bus and minimize the number of universal product terms required for each logic function. Within a quadrant, the I/O pins and associated logic are assigned first, and then any remaining buried resources can be assigned.

In this application, the address and data bus interfaces and the address and word counters would ideally be located in the same quadrant. This would minimize the universal bus routing and allow the counters to use buried registers. Since there are not enough resources in each quadrant to fit that much logic, the functions must be divided into smaller pieces. Figure 10 shows how the logic for this application is divided and allocated into quadrants.

The buses and counters are "bit-sliced" to create logic blocks which fit into each quadrant. The counters are divided by creating stages with a look-ahead carry in between. The first quadrant contains five bits of the data and address buses, five bits of each of the counters, and a carry bit from each counter. The second quadrant contains the next five-bit section and the third quadrant contains the last six-bit section. The carry bits are routed on the universal bus to the next quadrant. The state machine and command register are allocated to the fourth quadrant. The internal control signals generated by the state machine are routed on the universal bus to all quadrants.

Each of the counters would normally require three universal product terms for the load and count functions, since the control signals are universal. This would mean that each of the counters needs two sum terms in each logic cell. To avoid this, a regional load control signal was created in each quadrant for each of the counters, using the buried logic cells. Two of the counters universal product terms then become regional product terms, and the counters only require one sum term.

Each bit of the address bus is assigned to a combinatorial I/O pin. Each of those logic cells has two buried registers available, which are used for the address counters. The logic cell configuration is shown in Figure 4. Each bit of the data bus is assigned to a registered I/O pin. Each of those logic cells has an additional buried register available, which is used for the word counter. The carry bits for the counters are assigned to registered I/O pins so they can be routed universally.

The four state bits required seven, ten, four, and four product terms. In order to provide seven product terms, the first state bit was assigned to a registered I/O pin and allocated two sum terms. The other available register in that logic cell was assigned to one of the state bits which only required four product terms. The logic cell configuration is shown in Figure 5. The state bit which required ten product terms was assigned to a registered I/O pin and allocated all three sum terms. The control signals from the state machine (both internal and external) were assigned to registered I/O cells. The other available buried registers in those logic cells were assigned to the remaining state bit and the command register bits.

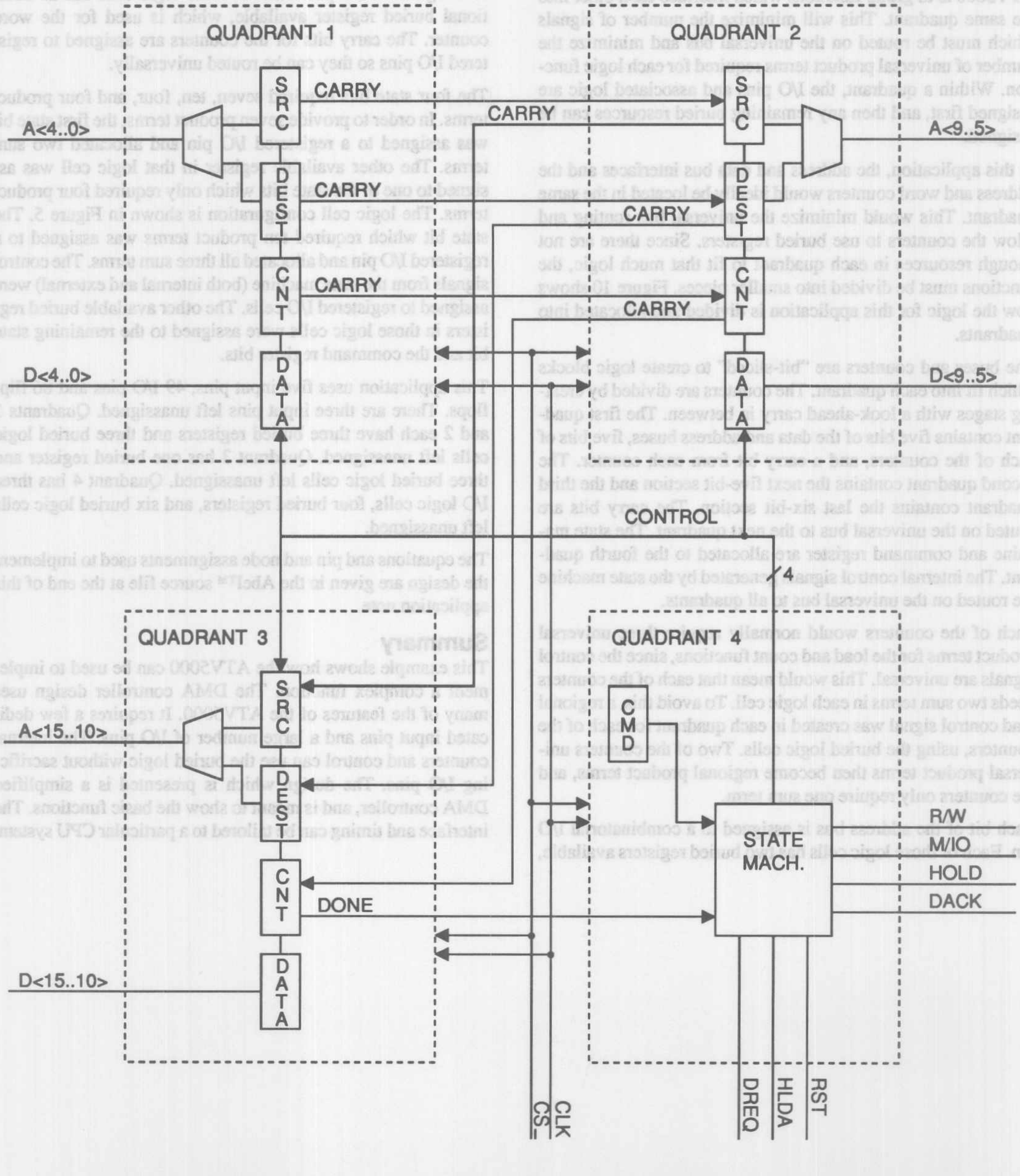
This application uses five input pins, 49 I/O pins and 86 flip-flops. There are three input pins left unassigned. Quadrants 1 and 2 each have three buried registers and three buried logic cells left unassigned. Quadrant 3 has one buried register and three buried logic cells left unassigned. Quadrant 4 has three I/O logic cells, four buried registers, and six buried logic cells left unassigned.

The equations and pin and node assignments used to implement the design are given in the Abel™ source file at the end of this application note.

Summary

This example shows how the ATV5000 can be used to implement a complex function. The DMA controller design uses many of the features of the ATV5000. It requires a few dedicated input pins and a large number of I/O pins. The internal counters and control can use the buried logic without sacrificing I/O pins. The design which is presented is a simplified DMA controller, and is meant to show the basic functions. The interface and timing can be tailored to a particular CPU system.

Figure 10. Resource Allocation



Abel™ Source Code

```
module DMAC ;
title 'DMA Controller - Example for the ATV5000
Atmel Corporation PLD - (408)436-4333 PLD Applications Hotline
Wendey Mueller - Sept. 15, 1992'

DMAC device 'P5000';

CLK,CS_,RST              pin 1,2,32;
HLDA,DREQ                pin 34,35;
D0,D1,D2,D3              pin 4,5,6,7 istype 'reg_d,buffer';
D4,D5,D6,D7              pin 8,18,19,21 istype 'reg_d,buffer';
D8,D9,D10,D11            pin 22,23,38,39 istype 'reg_d,buffer';
D12,D13,D14,D15          pin 40,41,42,43 istype 'reg_d,buffer';
A0,A1,A2,A3              pin 9,10,11,12 istype 'com,buffer';
A4,A5,A6,A7              pin 13,24,25,26 istype 'com,buffer';
A8,A9,A10,A11            pin 27,28,44,45 istype 'com,buffer';
A12,A13,A14,A15          pin 46,47,48,49 istype 'com,buffer';
R_W,M_IO                 pin 55,56 istype 'reg_d,buffer';
HOLD,DACK                pin 57,58 istype 'reg_d,buffer';

ST0,ST1                  pin 52,53 istype 'reg_d,buffer';
ST2,ST3                  node 160,162 istype 'reg_d,buffer';
EN,SM_IO,DM_IO           node 163,164,165 istype 'reg_d,buffer';
DMA_EN,BUS_OE            pin 59,60 istype 'reg_d,buffer';
CNTEN,LATEN              pin 61,62 istype 'reg_d,buffer';
CNT0,CNT1,CNT2,CNT3      node 121,122,123,124 istype 'reg_t,buffer';
CNT4,CNT5,CNT6,CNT7      node 125,134,135,136 istype 'reg_t,buffer';
CNT8,CNT9,CNT10,CNT11    node 137,138,147,148 istype 'reg_t,buffer';
CNT12,CNT13,CNT14,CNT15  node 149,150,151,152 istype 'reg_t,buffer';
SA0,SA1,SA2,SA3          node 766,767,768,769 istype 'reg_t,buffer';
SA4,SA5,SA6,SA7          node 770,779,780,781 istype 'reg_t,buffer';
SA8,SA9,SA10,SA11        node 782,783,793,794 istype 'reg_t,buffer';
SA12,SA13,SA14,SA15      node 795,796,797,798 istype 'reg_t,buffer';
DA0,DA1,DA2,DA3          node 126,127,128,129 istype 'reg_t,buffer';
DA4,DA5,DA6,DA7          node 130,139,140,141 istype 'reg_t,buffer';
DA8,DA9,DA10,DA11        node 142,143,153,154 istype 'reg_t,buffer';
DA12,DA13,DA14,DA15      node 155,156,157,158 istype 'reg_t,buffer';
SCARRY1,SCARRY2          pin 14,29 istype 'reg_d,buffer';
DCARRY1,DCARRY2          pin 15,30 istype 'reg_d,buffer';
CCARRY1,CCARRY2          pin 17,31 istype 'reg_d,buffer';
DONE                     pin 51 istype 'com,buffer';
LDSRC1,LDSRC2,LDSRC3     node 173,179,185 istype 'com,buffer';
LDDST1,LDDST2,LDDST3     node 174,180,186 istype 'com,buffer';
LDCNT1,LDCNT2,LDCNT3     node 175,181,187 istype 'com,buffer';

C,K,X,Z,U,D,H,L = .C.,.K.,.X.,.Z.,.U.,.D.,1,0;

"CREATE BUSES

DATABUS = [D15..D0];     "data bus
DATA = [D4..D0];  DATB = [D9..D5];  DATC = [D15..D10];
DLSB = [D7..D0];

ADDR = [A15..A0];        "address bus
ALSB = [A7..A0];

WRDCNT = [CNT15..CNT0]; "word count
CNTLSB = [CNT7..CNT0];
CNTA = [CNT4..CNT0];  CNTB = [CNT9..CNT5];  CNTC = [CNT15..CNT10];

SRCADR = [SA15..SA0];    "source address
SRCA = [SA4..SA0];  SRCB = [SA9..SA5];  SRCC = [SA15..SA10];
```

```
SRCLSB = [SA7..SA0];

DSTADR = [DA15..DA0];    "destination address
DSTA = [DA4..DA0];  DSTB = [DA9..DA5];  DSTC = [DA15..DA10];
DSTLSB = [DA7..DA0];

COMMAND = [EN,DM_IO,SM_IO]; "command register

STMACH = [ST3,ST2,ST1,ST0]; "state bits
STDATA = [CNTEN,LATEN,DMA_EN,BUS_OE,M_IO,R_W,DACK,HOLD]; "state machine outputs

"DEFINE STATES

S0  = [0,0,0,0];
S1  = [0,0,0,1];
S2  = [0,0,1,0];
S3  = [0,0,1,1];
S4  = [0,1,0,0];
S5  = [0,1,0,1];
S6  = [0,1,1,0];
S7  = [0,1,1,1];
S8  = [1,0,0,0];
S9  = [1,0,0,1];
S10 = [1,0,1,0];
S11 = [1,0,1,1];

"Register select for initialization
"
" A1  A0  register selected
"---------------------------
" 0   0   Command register
" 0   1   Source address register
" 1   0   Destination address register
" 1   1   Word count register
"

EQUATIONS

M_IO.OE = DMA_EN;  "M_IO and R_W are control outputs during DMA
R_W.OE = DMA_EN;

ADDR = (!BUS_OE & SRCADR.FB)     "Select source or destination address
     # (BUS_OE & DSTADR.FB);
ADDR.OE = DMA_EN;                "Address output during DMA

DATABUS.D = (DATABUS & LATEN)       "Clock in data on input bus if
          # (DATABUS.FB & !LATEN); "LATEN is true
DATABUS.CK = !CLK;
DATABUS.OE = BUS_OE;             "Data output to destination during DMA

COMMAND.D = [D2,D1,D0];          "load command register
COMMAND.CK = CLK & !CS_ & R_W & !A1 & !A0;
COMMAND.AR = DONE;

LDSRC1 = !CS_ & R_W & !A1 & A0; "create regional controls for
LDSRC2 = !CS_ & R_W & !A1 & A0; "source address counter
LDSRC3 = !CS_ & R_W & !A1 & A0;

LDDST1 = !CS_ & R_W & A1 & !A0; "create regional controls for
LDDST2 = !CS_ & R_W & A1 & !A0; "destination address counter
LDDST3 = !CS_ & R_W & A1 & !A0;

LDCNT1 = !CS_ & R_W & A1 & A0;  "create regional load controls for
LDCNT2 = !CS_ & R_W & A1 & A0;  "word counter
```

```
LDCNT3 = !CS_ & R_W & A1 & A0;

"Source address counter - 16-bit up-counter w/ parallel load
"                          3 stages with look-ahead carry in between

SRCC.T = ((SRCC.FB + 1) $ SRCC.FB) & CNTEN & SCARRY2 & SCARRY1 "count
       # (DATC.FB $ SRCC.FB) & LDSRC3;                         "load
SRCB.T = ((SRCB.FB + 1) $ SRCB.FB) & CNTEN & SCARRY1           "count
       # (DATB.FB $ SRCB.FB) & LDSRC2;                         "load
SRCA.T = ((SRCA.FB + 1) $ SRCA.FB) & CNTEN                     "count
       # (DATA.FB $ SRCA.FB) & LDSRC1;                         "load
SRCADR.CK = CLK;

SCARRY1.D = (SRCA.FB == ^h1E) & CNTEN  "source address 1st stage carry
          # SCARRY1.FB & !CNTEN;
SCARRY1.CK = CLK;
SCARRY2.D = (SRCB.FB == ^h1F) & CNTEN  "source address 2nd stage carry
          # SCARRY2.FB & !CNTEN;
SCARRY2.CK = CLK;

"Destination address counter - 16-bit up-counter w/ parallel load
"                               3 stages with look-ahead carry in between

DSTC.T = ((DSTC.FB + 1) $ DSTC.FB) & CNTEN & DCARRY2 & DCARRY1 "count
       # (DATC.FB $ DSTC.FB) & LDDST3;                         "load
DSTB.T = ((DSTB.FB + 1) $ DSTB.FB) & CNTEN & DCARRY1           "count
       # (DATB.FB $ DSTB.FB) & LDDST2;                         "load
DSTA.T = ((DSTA.FB + 1) $ DSTA.FB) & CNTEN                     "count
       # (DATA.FB $ DSTA.FB) & LDDST1;                         "load
DSTADR.CK = CLK;

DCARRY1.D = (DSTA.FB == ^h1E) & CNTEN  "destination address 1st stage carry
          # DCARRY1.FB & !CNTEN;
DCARRY1.CK = CLK;
DCARRY2.D = (DSTB.FB == ^h1F) & CNTEN  "destination address 2nd stage carry
          # DCARRY2.FB & !CNTEN;
DCARRY2.CK = CLK;

"Word counter - 16-bit down-counter w/ parallel load
"                3-stages with look-ahead carry in between

CNTC.T = ((CNTC.FB - 1) $ CNTC.FB) & CNTEN & CCARRY2 & CCARRY1 "count
       # (DATC.FB $ CNTC.FB) & LDCNT3;                         "load
CNTB.T = ((CNTB.FB - 1) $ CNTB.FB) & CNTEN & CCARRY1           "count
       # (DATB.FB $ CNTB.FB) & LDCNT2;                         "load
CNTA.T = ((CNTA.FB - 1) $ CNTA.FB) & CNTEN                     "count
       # (DATA.FB $ CNTA.FB) & LDCNT1;                         "load
WRDCNT.CK = CLK;

CCARRY1.D = (CNTA.FB == ^h01) & CNTEN  "word count 1st stage carry
          # CCARRY1.FB & !CNTEN;
CCARRY1.CK = CLK;
CCARRY2.D = (CNTB.FB == ^h00) & CNTEN  "word count 2nd stage carry
          # CCARRY2.FB & !CNTEN;
CCARRY2.CK = CLK;

DONE = (CNTC.FB == 0) & CCARRY1 & CCARRY2; "Detects when DMA is complete

STMACH.CK = CLK;
STDATA.CK = CLK;

"State Machine Controller -
"
"Inputs: RST,EN,SM_IO,DM_IO,DREQ,HLDA,DONE
```

```
"Outputs: CNTEN,LATEN,DMA_EN,BUS_OE,M_IO,R_W,DACK,HOLD

STATE_DIAGRAM STMACH

STATE S0:                        "Reset, idle state
   IF (EN & !RST) THEN           "If DMA is enabled, go to state 1
      S1 WITH STDATA.D = [0,1,0,0,0,0,0,0]; ENDWITH;
   ELSE                          "else, wait for enable
      S0 WITH STDATA.D = [0,1,0,0,0,0,0,0]; ENDWITH;

STATE S1:                        "DMA enabled, wait for DMA request
   IF (!EN # RST) THEN           "If disabled or reset, return to state 0
      S0 WITH STDATA.D = [0,1,0,0,0,0,0,0]; ENDWITH;
   ELSE IF (DREQ) THEN           "If DMA requested, go to state 2
      S2 WITH STDATA.D = [0,1,0,0,0,0,0,1]; ENDWITH;
   ELSE                          "else, wait for DMA request
      S1 WITH STDATA.D = [0,1,0,0,0,0,0,0]; ENDWITH;

STATE S2:                        "DMA requested, issue bus request
   IF (!EN # RST) THEN           "If disabled or reset, return to state 0
      S0 WITH STDATA.D = [0,1,0,0,0,0,0,0]; ENDWITH;
   ELSE IF (!DREQ) THEN          "If DMA request was removed, return to state 1
      S1 WITH STDATA.D = [0,1,0,0,0,0,0,0]; ENDWITH;
   ELSE IF (HLDA) THEN           "If bus control granted, go to state 3
      S3 WITH STDATA.D = [0,1,0,0,0,0,1,1]; ENDWITH;
   ELSE                          "else, wait for bus control
      S2 WITH STDATA.D = [0,1,0,0,0,0,0,1]; ENDWITH;

STATE S3:                        "Bus granted, issue DMA acknowledge
   IF (!EN # RST) THEN           "If disabled or reset, return to state 0
      S0 WITH STDATA.D = [0,1,0,0,0,0,0,0]; ENDWITH;
   ELSE IF (!DREQ) THEN          "If DMA request was removed, return to state 1
      S1 WITH STDATA.D = [0,1,0,0,0,0,0,0]; ENDWITH;
   ELSE IF (!HLDA) THEN          "If bus control was revoked, return to state 2
      S2 WITH STDATA.D = [0,1,0,0,0,0,0,1]; ENDWITH;
   ELSE IF (SM_IO & !DM_IO) THEN  "Start DMA: Memory - I/O Device
      S4 WITH STDATA.D = [0,1,1,0,1,0,1,1]; ENDWITH;
   ELSE IF (!SM_IO & DM_IO) THEN  "Start DMA: I/O Device - Memory
      S6 WITH STDATA.D = [0,1,1,0,0,0,1,1]; ENDWITH;
   ELSE IF (SM_IO & DM_IO) THEN   "Start DMA: Memory - Memory
      S8 WITH STDATA.D = [0,1,1,0,1,0,1,1]; ENDWITH;
   ELSE IF (!SM_IO & !DM_IO) THEN "Start DMA: I/O Device - I/O Device
      S10 WITH STDATA.D = [0,1,1,0,0,0,1,1]; ENDWITH;

STATE S4:                        "DMA cycle, memory read
   IF (!DREQ # RST) THEN         "If DMA request removed or reset, return to state 0
      S0 WITH STDATA.D = [0,1,0,0,0,0,0,0]; ENDWITH;
   ELSE                          "else, continue DMA transfer
      S5 WITH STDATA.D = [1,0,1,1,0,1,1,1]; ENDWITH;

STATE S5:                        "DMA cycle, I/O device write
   IF (!DREQ # RST # DONE) THEN   "If DMA request removed, reset or done,
                                 "return to state 0
      S0 WITH STDATA.D = [0,1,0,0,0,0,0,0]; ENDWITH;
   ELSE IF (!HLDA) THEN           "If bus control revoked, return to state 2
      S2 WITH STDATA.D = [0,1,0,0,0,0,0,1]; ENDWITH;
   ELSE                           "else, continue DMA transfer
      S4 WITH STDATA.D = [0,1,1,0,1,0,1,1]; ENDWITH;

STATE S6:                        "DMA cycle, I/O device read
   IF (!DREQ # RST) THEN         "If DMA request removed or reset, return to state 0
      S0 WITH STDATA.D = [0,1,0,0,0,0,0,0]; ENDWITH;
   ELSE                          "else, continue DMA transfer
      S7 WITH STDATA.D = [1,0,1,1,1,1,1,1]; ENDWITH;
```

```
STATE S7:                        "DMA cycle, memory write
   IF (!DREQ # RST # DONE) THEN  "If DMA request removed, reset or done,
                                 "return to state 0
      S0 WITH STDATA.D = [0,1,0,0,0,0,0,0]; ENDWITH;
   ELSE IF (!HLDA) THEN          "If bus control revoked, return to state 2
      S2 WITH STDATA.D = [0,1,0,0,0,0,0,1]; ENDWITH;
   ELSE                          "else, continue DMA transfer
      S6 WITH STDATA.D = [0,1,1,0,0,0,1,1]; ENDWITH;

STATE S8:                        "DMA cycle, memory read
   IF (!DREQ # RST) THEN         "If DMA request removed or reset, return to state 0
      S0 WITH STDATA.D = [0,1,0,0,0,0,0,0]; ENDWITH;
   ELSE                          "else, continue DMA transfer
      S9 WITH STDATA.D = [1,0,1,1,1,1,1,1]; ENDWITH;

STATE S9:                        "DMA cycle, memory write
   IF (!DREQ # RST # DONE) THEN "If DMA request removed, reset or done,
                                 "return to state 0
      S0 WITH STDATA.D = [0,1,0,0,0,0,0,0]; ENDWITH;
   ELSE IF (!HLDA) THEN          "If bus control revoked, return to state 2
      S2 WITH STDATA.D = [0,1,0,0,0,0,0,1]; ENDWITH;
   ELSE                          "else, continue DMA transfer
      S8 WITH STDATA.D = [0,1,1,0,1,0,1,1]; ENDWITH;

STATE S10:                       "DMA cycle, I/O device read
   IF (!DREQ # RST) THEN         "If DMA request removed or reset, return to state 0
      S0 WITH STDATA.D = [0,1,0,0,0,0,0,0]; ENDWITH;
   ELSE                          "else, continue DMA transfer
      S11 WITH STDATA.D = [1,0,1,1,0,1,1,1]; ENDWITH;

STATE S11:                       "DMA cycle, I/O device write
   IF (!DREQ # RST # DONE) THEN  "If DMA request removed, reset or done,
                                 "return to state 0
      S0 WITH STDATA.D = [0,1,0,0,0,0,0,0]; ENDWITH;
   ELSE IF (!HLDA) THEN          "If bus control revoked, return to state 2
      S2 WITH STDATA.D = [0,1,0,0,0,0,0,1]; ENDWITH;
   ELSE                          "else, continue DMA transfer
      S10 WITH STDATA.D = [0,1,1,0,0,0,1,1]; ENDWITH;

"This set of test vectors initializes the source address, destination address,
"word count and command data.

TEST_VECTORS (
[CLK,RST,CS_,M_IO,R_W,ALSB,DLSB] - [SRCLSB,DSTLSB,CNTLSB,COMMAND])
[ 1,  0, 1,  0,   1,  0,   0   ] - [ 0,     0,     0,    0      ];
[ 1,  1, 1,  0,   1,  0,   0   ] - [ 0,     0,     0,    0      ];
[ K,  0, 0,  0,   1,  1,   00A ] - [00A,    0,     0,    0      ];
[ K,  0, 0,  0,   1,  2,   0B0 ] - [00A,   0B0,    0,    0      ];
[ K,  0, 0,  0,   1,  3,   003 ] - [00A,   0B0,   003,   0      ];
[ K,  0, 0,  0,   1,  0,   007 ] - [00A,   0B0,   003,   7      ];

"This set of test vectors performs a DMA transfer using the addresses and word
"count which was loaded.

TEST_VECTORS (
[CLK,RST,CS_,ALSB,DLSB,DREQ,HLDA] - [STMACH,STDATA,SRCLSB,DSTLSB,CNTLSB,DONE])
[ 1, 0,  1,  0,   0,   0,   0   ] - [1,     X,     0A,    0B0,   03,    0   ];
[ C, 0,  1,  0,   0,   0,   0   ] - [1,     X,     0A,    0B0,   03,    0   ];
[ C, 0,  1,  0,   0,   1,   0   ] - [2,     X,     0A,    0B0,   03,    0   ];
[ C, 0,  1,  0,   0,   1,   1   ] - [3,     X,     0A,    0B0,   03,    0   ];
[ C, 0,  1,  0,   55,  1,   1   ] - [8,     06B,   0A,    0B0,   03,    0   ];
[ C, 0,  1,  0,   0,   1,   1   ] - [9,     0BF,   0A,    0B0,   03,    0   ];
[ C, 0,  1,  0,   66,  1,   1   ] - [8,     06B,   0B,    0B1,   02,    0   ];
```

5

```
[ C, 0,  1,  0,   0,  1,   1  ] - [9,     0BF,   0B,    0B1,   02,    0   ];
[ C, 0,  1,  0,  77,  1,   1  ] - [8,     06B,   0C,    0B2,   01,    0   ];
[ C, 0,  1,  0,   0,  1,   1  ] - [9,     0BF,   0C,    0B2,   01,    0   ];
[ C, 0,  1,  0,   0,  1,   1  ] - [8,     06B,   0D,    0B3,   00,    1   ];
[ C, 0,  1,  0,   0,  1,   1  ] - [9,     0BF,   0D,    0B3,   00,    1   ];
[ C, 0,  1,  0,   0,  1,   1  ] - [0,     X,     0E,    0B4,   0FF,   0   ];

END ;
```

Contents

Section 6 FPGA and Gate Array Application Notes

Recommended Design Methods

Field Programmable Gate Array

Application Note

6

Introduction

Described here are a series of guidelines for designing with AT6000 Series field programmable gate arrays (FPGAs). Among the topics covered are basic cell functionality, building simple functions, general manual placement-and-routing rules, and schematic-entry tips that can make time spent in the Interactive Editor more productive. Keeping these guidelines in mind can reduce design time and produce more efficient circuits.

The Basics

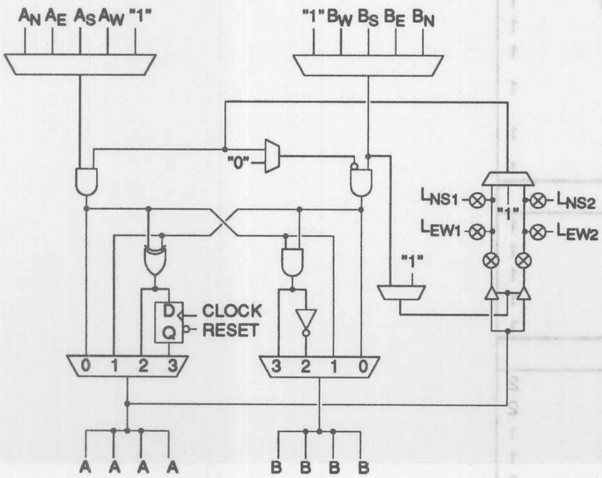
Before beginning a design, it is important to understand the building blocks of the Atmel architecture. AT6000 Series devices consist of a symmetrical array of cells that perform logic functions and are connected to a comprehensive busing structure. Symmetrical interconnects on the four sides of each cell provide cell-to-cell and cell-to-bus connections. Figure 1 shows the logic contained in each cell. A more detailed explanation of cell functionality can be found in the AT6000 Series data sheet.

Logical Functions and Cell Configurations

The logical function is the function a cell is performing; the cell configuration is how the cell is configured to perform that function. Cells can perform 44 logical functions, each corresponding to at least one cell configuration.

For example, a cell can be configured six different ways to perform the same inverter function. Figures 2a and 2b show two of these configurations. The inverter in 2a goes from a local bus input (L) to a B output. The inverter in 2b takes an A input signal, inverts it, and drives it to the A, B, and L outputs. Depending on routing conditions and the use of neighboring cells, one configuration may be more appropriate than another.

Figure 1. Cell Structure



The following table shows the number of cell configurations associated with each logical function:

Logical Function		Cell Configurations
Combinatorial (1 output)		
INV	Inverter	6
AN2	2-input AND	4
ND2	2-input NAND	4
XO2	2-input XOR	2
ORT	2-input OR	1
MUX	A = (A•L) XOR (B•L')	1
AN2L	(B•L') 2-input AND with 1 Inverted Input	1
ORL	(A+L') 2-input OR with 1 Inverted Input	1
AN3	3-input AND	1
ND3	3-input NAND	1
ANXO	2-input AND Feeding an XOR	1
BUF	Buffer	1
Combinatorial (2 outputs)		
INVW	Local Bus Input Inverter with Thru Wire	2
INVINV	Two Inverters	3
AN2S	2-input AND and B Wire	1
AN2X	2-input AND and B Cross Wire	1
AN2INV	(A•L,L') 2-input AND and Inverter	1
INVAN2	(L',A•L) Inverter and 2-input AND	1
NDND	Two 2-input NANDS	1
XOND	2-input XOR & NAND	2
SELBUFS	A = (A•L);B = (B•L')	1
SELBUFX	A = (B•L');B = (A•L)	2
XOND3	A = (A•L) XOR B;B = (A•L) NAND B	1
WAN2L	Wire & (B•L')	1
AN2LW	(B•L') & Wire	1
Flip-Flop		
FD	D Flip-Flop Q Out	1
FDN	D Flip-Flop QN Out	3
FDHA	D Flip-Flop = Half-Adder Sum	2
FDMUX	D Flip-Flop = (A•L) XOR (B•L')	1
FDND	D Flip-Flop = 2-input NAND	1
FDOR	D Flip-Flop = 2-input OR	1
FDORL	D Flip-Flop = 2-input OR with 1 Inverted Input	1
FDXOAN3	D Flip-Flop = (A•L) XOR B; B = (A•L) AND B	1
CLKEDGE	Clock Edge Detect	1
Tri-state		
BUFZ	Tri-state Buffer	1
FDZ	Tri-state D Flip-Flop	1
LZ	"0" or "Z" (high impedance)	1
HZ	"1" or "Z" (high impedance)	1
CLKEDGEZ	Clock Edge Detect or "Z"	1
Constant		
ONE	Logic One	2
ZERO	Logic Zero	2
ONEONE	Two Logic Ones	1
ZEROZERO	Two Logic Zeros	1
ZEROONE	Logic One and Logic Zero	2

Figure 2a. First Inverter Configuration

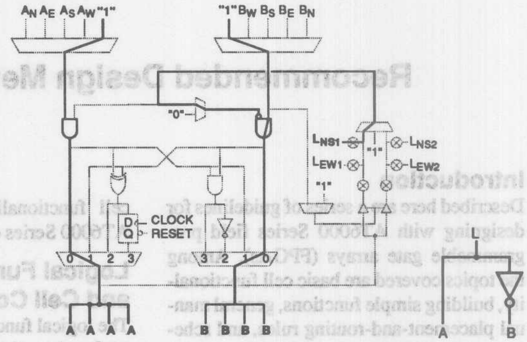
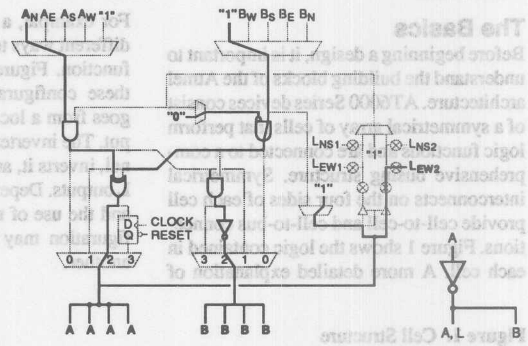


Figure 2b. Second Inverter Configuration



Combining Cell Functions

When cells are used in combination, more functions are available. The combination of two cells, for example, can produce a four-input AND gate as shown in Figure 3. Other two-cell functions include a three-input XOR and a set/reset NAND latch.

The combination of three cells can produce more complex operations such as the two-input AND/OR function shown in Figure 4.

The three cells in combination produce a specific output, but the outputs of each cell can be used as part of another function. The following table lists the intermediate outputs available from this function:

Inputs		Outputs					
A	B	NAND	XOR	XNOR	NOR	OR	AND
0	0	1	0	1	1	0	0
0	1	1	1	0	0	1	0
1	0	1	1	0	0	1	0
1	1	0	0	1	0	1	1

This same two-input AND/OR function could be implemented using a single cell, but that would preclude use of the intermediate outputs.

Macros

When cells are grouped together to perform a specific logical function, they form a macro. A single-macro function, like a single logical function, can be configured in more than one way. Each of these physical variations is called a shape. Shapes vary in their use of routing resources, the relative placement of logic cells, and the number and type of physical primitives used. Routing varies from shape to shape, making some shapes faster than others. Designs can therefore be tuned for speed or size.

Macros are either hard or soft. Hard macros maintain the relative placement of each logic cell. The timing of a hard macro can be fully characterized and remains constant regardless of the macro's orientation in the layout because the relative placement of its components remains unchanged. Because the AT6000 Series architecture is symmetrical, hard macros can be flipped or rotated without affecting internal macro timing.

Soft macros are made by dismantling, or softening, a hard macro, or by creating a schematic symbol that represents a series of hard macros. (In essence, every unplaced design is really a large, soft macro.) Each cell in a soft macro can be placed individually, so the timing of a soft macro depends on cell placement and interconnect.

A list of the more than 200 macros included in the Atmel Macro Library appears in the Integrated Development System data sheet. Most macros have more than one shape, yielding a total of over 350 configurations.

Macros can be combined to make more complex functions. For example, six half-adders (HA1, shown in Figure 5a) can combine to create the 4-bit Summer shown in Figure 5b. The Summer uses four binary inputs and defines its outputs as follows:

If 0 bits = 1, then output = 0 (Binary 000)

If 1 bit = 1, then output = 1 (Binary 001)

If 2 bits = 1, then output = 2 (Binary 010)

If 3 bits = 1, then output = 3 (Binary 011)

If 4 bits = 1, then output = 4 (Binary 100)

The Summer counts how many of the four-input bits are true and outputs the sum. Figure 5c shows the layout of the Summer in the Atmel architecture.

Figure 3. Two Cells Combine to Make a Four-Input AND

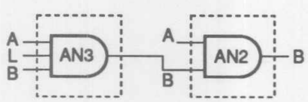
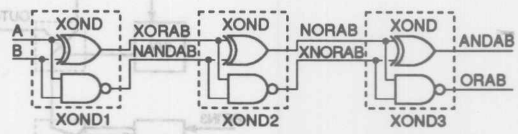


Figure 4. A Two-Input AND/OR Functions Uses Three Cells



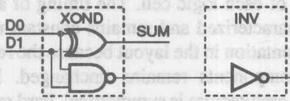


Figure 5b. Six HA1 Macros Used to Build a 4-Bit Summer Function

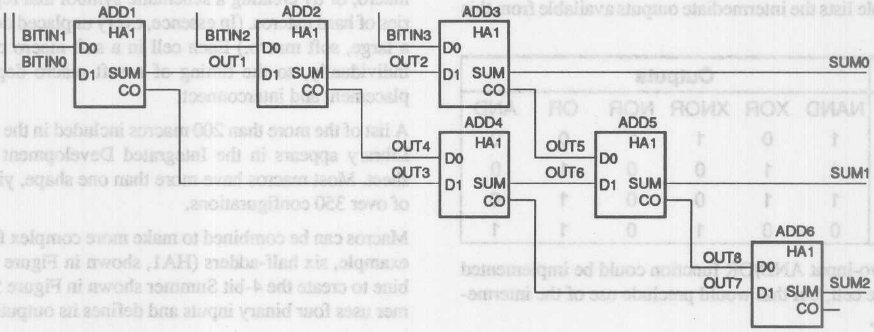
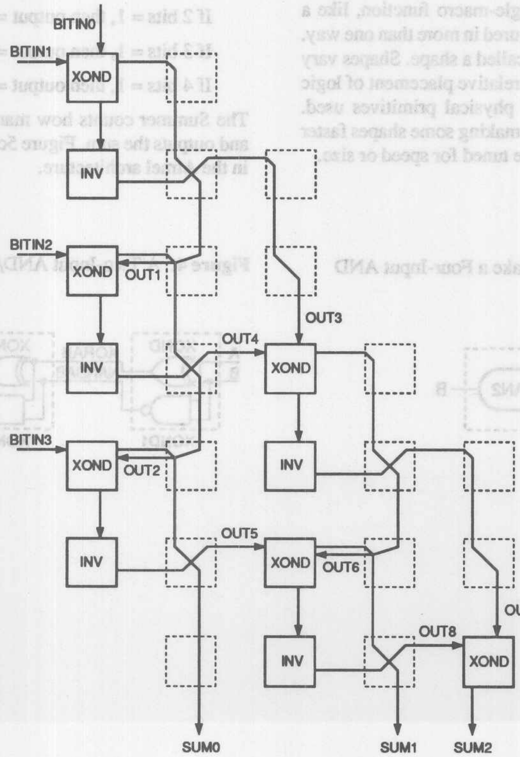


Figure 5c. Layout of the 4-Bit Summer



Placement and Routing

Once the macros to be used in a design have been selected and the schematic is complete, the design *is* ready to be placed and routed. Placement and routing are interdependent because cells can be used for both logic and routing. The relationship between placement and routing creates a number of trade-offs involving circuit speed, cell utilization and location of routed nets. For example, cell placement can interfere with routing by blocking a cell and making the net unroutable. Understanding these trade-offs makes place and route easier and more efficient.

Routing Rule 1: Conserve busing resources.

In general, because cells are more plentiful than buses, cells should be used in place of buses for routing signals over short distances. Placing interconnected cells in adjacent locations in the array, or abutting them, avoids routing delays and makes it easy to route the signals together.

To conserve buses, it is sometimes better to use more than the minimum number of cells to implement a function. For example, a two-input OR gate can be performed in one cell, but requires the use of a local bus, as shown in Figure 6a. If the local bus is already in use, then it is better to implement the function using three cells and no local bus, as in Figure 6b. Using a bus-free macro implementation makes layout more flexible because the macro can be placed and routed with less restriction.

Busing resources are best saved for routing signals across longer distances (more than five cells) in the array, for tri-state capabilities, for signals with high fanout, and for hard-to-route nets.

Routing Rule 2: Align common signals used as inputs.

When a number of signals provide the inputs for many functions, group the signals together and route them in parallel.

Consider implementing the decoder function shown in Figure 7a. The decoder implements full or partial product terms:

Q1 = S3' & S2 & S1 & S0'

Q2 = S3 & S2' & S1' & S0'

Q3 = S3 & S0'

The layout of the decoder is shown in Figure 7b. Each cell receives the same input via a local bus, and directs the input to an inverter (INV), a two-input AND gate (AN2), or a two-input gate with one inverted input (AN2L). Because Q1, Q2, and Q3 go through the same number of cells, timing for each signal is regular.

Consider also the two-to-four decoder shown in Figure 8a. The logic schematic and physical layout (Figure 8b) show this function implemented using four cells and two local buses. As in the product term example above, the two inputs S1 and S0 enter the cells from the local bus, but signals could enter from adjacent cells if the local bus was already being used to route another signal. The outputs exit from the A and B cell outputs.

Figure 6a. Two-Input OR Using One Cell and a Local Bus

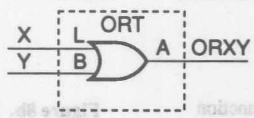


Figure 6b. Two-Input OR Using Three Cells and No Bus

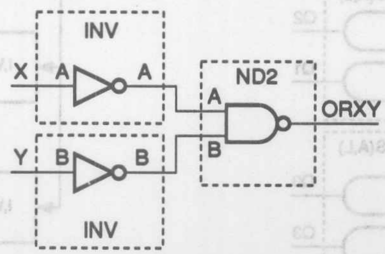


Figure 7a. Schematic of Decoder Function

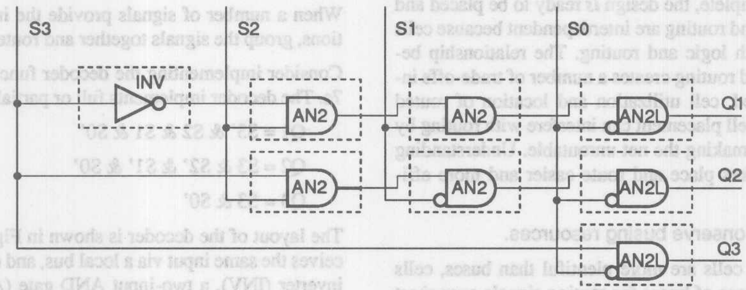


Figure 7b. Layout of the Decoder

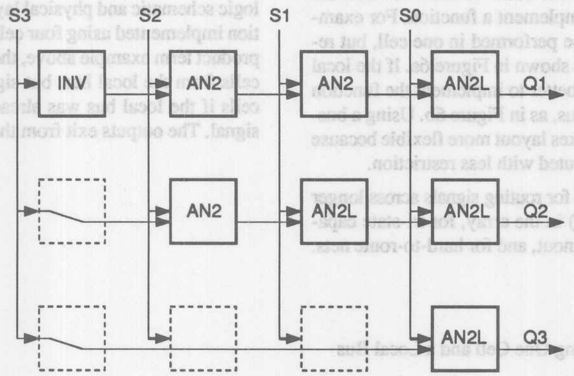


Figure 8a. Schematic of Two-to-Four Decoder Function

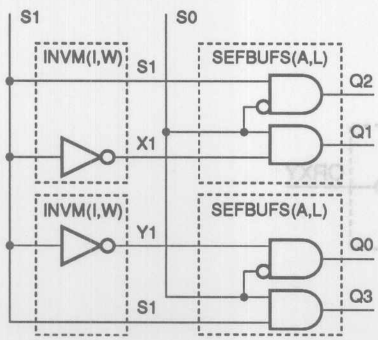
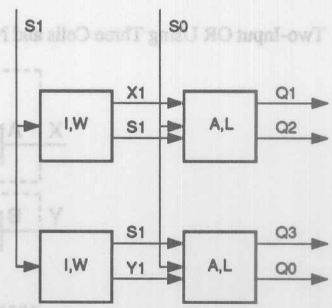


Figure 8b. Layout of Two-to-Four Decoder Function



Routing Rule 3: Use express buses whenever possible.

Because they are not connected directly to cells and thus have lower capacitive loads, express buses are faster than local buses and should be used whenever possible to increase design performance. Also, using an express bus in place of a local bus frees up the local bus for other routing purposes. In some cases, however, substituting an express for a local bus will not be possible:

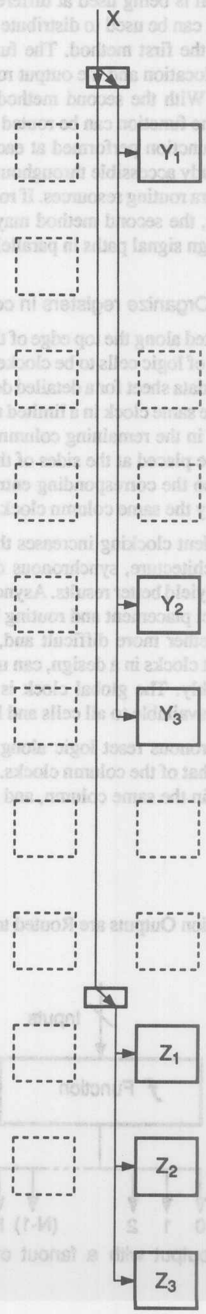
- when connecting directly to a cell
- when using a bi-directional signal
- when making 90° turns

For increased performance it is best to limit the number of local bus segments carrying a signal and to cross repeater boundaries only if necessary.

Branching the express bus signal to the local bus at each repeater can be beneficial when the fanout of a signal is greater than eight or the signal goes through more than one repeater. This helps balance the load of each local bus segment, making timing consistent across the length of the array.

Figure 9 illustrates an example of branching. Signal X is routed through the express bus and branches at the repeaters to the local bus segments driving the Y and Z cells. If signal X had been routed via the local bus to cells Y_1, Y_2, and Y_3, and through a repeater (local bus to local bus) to cells Z_1, Z_2, and Z_3, the load of the Y cells would impact the speed of the signal reaching the Z cells.

Figure 9. Branching Signals Increases Performance



If the same function is being used at different locations in the array, two methods can be used to distribute the functions. Figure 10a illustrates the first method. The function can be performed at a single location and the output routed to other locations in the array. With the second method, shown in Figure 10b, the inputs of the function can be routed to each location of the array and the function performed at each location. Unless the signals are already accessible throughout the array, this option can require extra routing resources. If routing resources are available, however, the second method may be preferable because it lets you align signal paths in parallel and makes timing more consistent.

Routing Rule 5: Organize registers in columns.

Clocks are distributed along the top edge of the Atmel array and allow each column of logic cells to be clocked individually (see the AT6000 Series data sheet for a detailed description). Grouping cells that use the same clock in a limited number of columns frees up the clocks in the remaining columns. Similarly, registered I/Os should be placed at the sides of the array rather than the top or bottom so the corresponding entrance and exit cells can be controlled by the same column clock.

Although independent clocking increases the flexibility of the AT6000 Series architecture, synchronous designs based on a single global clock yield better results. Asynchronously clocked registers can restrict placement and routing because they make grouping cells together more difficult and, depending on the number of different clocks in a design, can use up limited clock resources too quickly. The global clock is the most efficient clock because it is available to all cells and has low skew.

Use of the asynchronous reset logic along the bottom of the array is similar to that of the column clocks. Place logic requiring the same reset in the same column, and limit the variety of resets in a design.

Although it is possible to implement designs directly in the layout using the Interactive Editor, many engineers describe their designs in a schematic and use the resulting netlist for placement and routing. The tips included here describe how to label macros and nets in Viewdraw for use with the Interactive Editor in netlist-driven mode.

Labels applied in the schematic carry over to the "to-be-placed" list of macros in the Interactive Editor. The label from each hierarchical level is included in the list, beginning with the top-level symbol and continuing down the hierarchy to the macro label. For easy viewing in the placement menu, use a short label to identify each instance of a component on the schematic.

Component labels are used in the Interactive Editor for placement. Labels can be entered at the keyboard or selected from a placement menu. The placement menu lists labels in alphabetical order, with the first item preselected. Careful use of labels can organize macros such that functions appear together, making placement go more quickly.

Nets that connect components can also be labeled for easy identification, making it simple to correlate placement and routing with the design schematic. Workview automatically assigns names to unlabeled nets (see Workview manuals for details), so it is best to create meaningful labels as you create the schematic.

Making your own components

Viewlogic lets you create components by defining symbols to represent underlying schematics. The elements of the Viewlogic component appear in the Interactive Editor's placement list under a common root name. The underlying elements can be selected from the list and placed like any other Atmel macro.

The Design Manager automatically creates a symbol for the Atmel FPGA using the 132-pin PQFP package by default. This symbol can be modified to specify I/O connections, or replaced with any of the other package symbols.

Figure 10a. Function Outputs are Routed to Other Locations in the Array

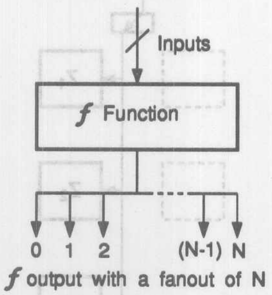
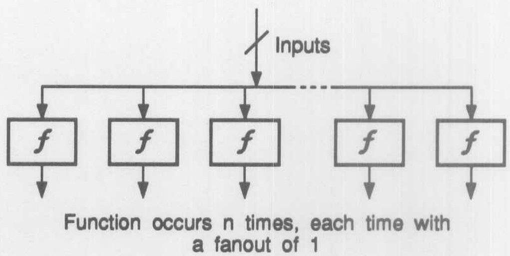


Figure 10b. The Function is Performed at Different Locations in the Array



Implementing Cache Logic™ with FPGAs

Field Programmable Gate Array

Application Note

by Joel Rosenberg

The Cache Logic Concept

Atmel Corporation has developed an enabling technology to make adaptive hardware possible for electronics systems. This capability, trademarked as *Cache Logic*, was developed and patented by Concurrent Logic (recently acquired by Atmel Corporation, Inc.).[1]

Cache logic is a cost-saving way of implementing logic more efficiently. The active functions of an application are performed by a field programmable gate array (FPGA) that can be reconfigured as it operates, while inactive functions are stored in an inexpensive configuration memory—an EPROM, for example. As new functions are required, they are written over old ones.

A single application is made up of many smaller macro-level operations, like counters, multipliers, shift registers, and multiplexers. When an application is broken down into its sub-operations, two things become apparent. First, functionality overlaps. A single function may be used a number of different times. Second, there is a high degree of functional latency. At any given moment, only a small portion of an application's operations are active; only a few functions are used at the same time.

By consolidating functionality, eliminating redundancy, and tracking the occurrence of each sub-operation, functions can be organized such that a relatively small, inexpensive logic device is reconfigured as it operates to perform a complex function. In a 10,000-gate application, for example, only 2,000 gates might be active at once. By caching the extra 8,000 gates for later use, a 2,000-gate device replaces a more expensive 10,000-gate device.

Cache Logic Implementation

Cache logic implementation is conceptually similar to cache memory. In cache memory, the highest speed memory (usually SRAM) is used to store active data, while the bulk of data resides in lower-cost storage, such as DRAM, or EPROM, disk, etc. Cache logic works in a similar fashion. Only a small fraction of the circuitry —those functions which are loaded into the logic cache—is active in a system at any given time, while unused functions or variations reside in lower-cost system memory. It is even possible to compile variations of a design in real time. As logic functions are required, they can be loaded into cache logic, replacing or complementing the logic already present.

Figure 1 shows the block diagram for the Atmel AT6000 FPGA, which is an ideal medium for cache logic. The ability to implement cache logic requires FPGAs that are capable of being dynamically reconfigured in system, either completely or partially, without disrupting the operation of the balance of logic in the device. Another requirement is architecture symmetry. This is necessary to make possible the arbitrary placement of generic blocks in a location that is available at the time required. It is also necessary to allow for easy modeling of device characteristics for the artificial intelligence required in the partitioning of a design. The symmetry also simplifies the creation of arrays of devices to create a larger digital medium for the implementation of cache logic.

Note: (1) The method for exploiting Cache Logic was pioneered by the University of Strathclyde in Scotland and is described in Lysaght, P. and Dunlop, J., "Dynamic Reconfiguration of Field Programmable Gate Arrays", in *More FPGAs*, W. Moore and W. Luk, Eds., Abingdon EE&CS Books, England 1994.

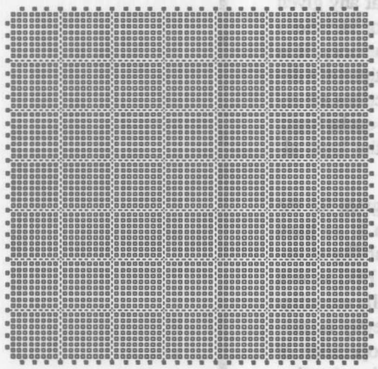
Predetermined and Dynamic Cache Logic

There are two types of cache logic which have been defined: *predetermined* cache logic and *dynamic* cache logic. *Predetermined* cache logic involves the use of predefined functions and macros that are stored in external, nonvolatile memory (EPROM, EEPROM, disk, CD ROM, or even memory remote from the system loaded over a communications link). These functions have already been placed and routed and have bit streams which have been previously generated (Figure 2). The implementation of these functions is controlled by a resident manager in the logic cache, or in an external control such as a microcontroller/processor routine. New functions may be downloaded to the logic cache in the background without disrupting the operation of the cache (logic, I/O, and register data), as shown in Figure 3. In fact, data in the registers is not lost even in the area being overwritten.

The second type of cache logic, *dynamic*, is the basis for building adaptive hardware. Dynamic caching involves the determination of logic, placement and routing of the logic, bit stream generation, and programming the logic cache in real time. The major issues to be addressed in the development of this capability include (but are not limited to) the scheduling and allocation of functions, random-logic collection, and collision handling and avoidance within the cache. Dynamic cache logic exists as a concept today; the physical implementation issues described above have not yet been fully addressed.

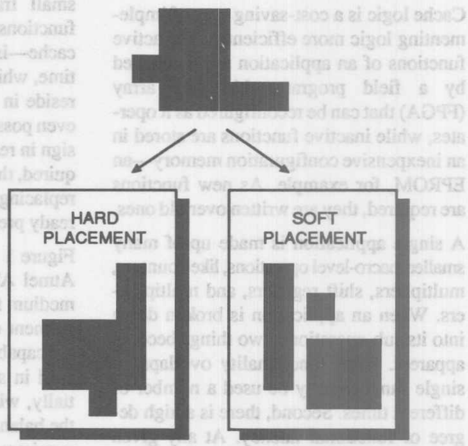
Cache logic may be applied in many applications. The concept of *virtual products* will be introduced, which utilizes the flexibility of programmable logic. Virtual products do not require cache logic programmability but, as we see, the use of cache logic greatly reduces the amount of programmable digital media needed to implement a virtual product.

Figure 1. AT6000 Array



- Symmetrical Array
  - Identical Cells
  - 8-by-8 Cell Sectors
  - Programmable Interconnects
  - Surrounded by I/O
- No Dedicated Functions
- Reconfigurable On-the-Fly
  - Full
  - Partial
  - Without Data Loss

Figure 2. Macro Library



- Over 200 Hard Macros
  - Fast
  - Fully Specified
  - Fixed Routing
- All Can be Softened
  - Flexible Placement
- User-Defined Macros
  - Create Own Library
  - Use on Future Designs
- Test Macros
  - For Debug/System Test
- Super Macros
  - Major Predefined Functions
  - Specialized for Markets

Virtual Products

A *virtual product* is a combination of a "tangible asset," such as a data acquisition board, and a service, such as product customization. The first thing to understand about virtual products is what the end customer wants, and how system developers can match their core competencies with these needs.

There are two issues raised in the manufacture of virtual products:

1. How to balance economy of scale achieved in volume manufacturing with special features that customers are willing to pay for; and
2. How to create diversity while maintaining a level of quality associated with standard high-volume production.

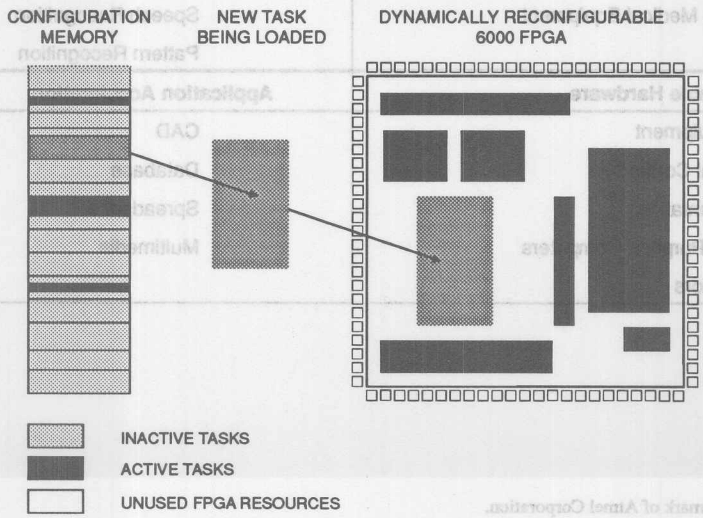
Cache Logic and FPGAs help the manufacturers achieve these two requirements of virtual products. A virtual product line is one with characteristics which meet the needs of a class of customers. An example would be a PC-based data-acquisition product. Such a product has certain physical requirements consistent with a PC-bus card standard. The board would also have a series of standard data gathering features such as multiple-channel A-to-D converters, digital I/O ports, D-to-A converters, and high-speed clock counters. These features are typically accomplished by highly integrated well-designed ICs readily available to all manufactures. The complexity of such products is in the data path and protocol which connects the PC to the standard IC products. The structure of this data path is prejudiced by optimum system performance, cost, and customer preference, the key item being customer preference for a successful virtual product, or for that matter any successful product.

The traditional approach to creating a data-acquisition product, like most products, is to create a board with a standard bus footprint, use industry standard A-to-D and D-to-A circuits, and then create a custom data path. The manufacturer then has to trust marketing studies and instinct to determine the best data path approach. It is possible to hedge the bet by adding redundancy. This redundancy has two detrimental effects: added cost, and added complexity for the end user. The selection of wrong data path protocol or excessive complexity caused by redundancy results in dissatisfied or nonexistent customers.

A virtual product does not mean that a manufacturer would be able to offer one product which was all things to all people. The use of programmable logic would allow a manufacturer to create an extensive catalog of products, but only have a small number of tangible assemblies to tool for manufacturing. The manufacturer would use FPGAs and cache logic FPGAs to create diversity in its product line. The cost of diversity to the manufacturer is the cost of service, or "personalization engineering," required to create a niche design on a standard assembly. The advantage for the customer is a mass-produced product which meets their specific needs.

The virtual product approach allows a manufacturer to perfect a single assembly. The FPGA's ability to be configured for self-test could even enhance the quality of the assembly. Atmel has developed the IEEE1148 boundary-scan supermacro. Utilizing the reconfigurable logic capability of the AT6000 family, the boundary-scan function may be loaded into the device and diagnostics performed, then the device can be reconfigured for other logic functions. A single Atmel device may be used for testing and logic, with no overhead or speed penalty, as is the case for all other FPGAs and other ASIC devices.

Figure 3. Cache Logic Concept



The result would be an inventory of nearly identical raw-product assemblies, which through virtual design becomes a catalog full of products when shipped to the customer. With a solid design, most customer problems can be traced to the virtual-design personalization process, and be repaired in the field with FPGA configuration updates. It is also possible to introduce new features into virtual products as soon as they are invented and proven, rather than wait until a new hardware product is designed, tooled, and manufactured.

Cache Logic Benefits

There are several benefits derived from cache logic design:

- New functionality may be added to existing hardware, without having to make modifications to the board.
- The hardware may be tailored to the application, resulting in higher system performance across a broad range of applications.
- The FPGA density limitations are eliminated.
- Overall system reliability is improved by reducing the number of physical products manufactured and utilizing boundary scan macros for manufacturing and system testing.
- Overall product life cycle costs are significantly reduced by using reusable software and hardware:
  - lower development costs
  - lower inventory costs
  - quicker time to market
  - fewer parts on the board
  - lower power consumption
  - lower total system cost
  - reusable designs

Summary

To many people, the ideal of adaptive hardware or virtual products is a futuristic concept. The AT6000 family is capable of implementing cache logic and virtual products today. The Atmel FPGA and its abilities to implement cache logic make it a foundation for adaptive hardware and virtual products. Successful design with this new technology has been commercially demonstrated. Today's design methodology, that requires a new product for each new function, will be replaced by adaptive hardware products that meet the needs of both customers and suppliers with customized products and improved quality, while reducing product development time and overall life cycle costs.

Table 1. Examples of Cache Logic™ Applications

Power and Space-Sensitive Applications	Compute Intensive Applications
Portable Computers	Computer Graphics
Battery-Operated Instrumentation	Image Processing
Portable Communications	Data Compression
Portable Medical Equipment	Speech Recognition
	Pattern Recognition
Reprogrammable Hardware	**Application Acceleration**
Test Equipment	CAD
Industrial Control	Database
Instrumentation	Spreadsheet
Special-Purpose Computers	Multimedia
Connectors	

Cache Logic™ is a trademark of Atmel Corporation.

Data Acquisition Systems Using Cache Logic™ FPGAs

Field Programmable Gate Array

Application Note

by Rafe Camerota, Design Manager and Joel Rosenberg, Marketing Manager

Atmel has developed an enabling technology to make adaptive hardware possible for Data Acquisition, Logic Analyzer, and other instrumentation products. This capability, trademarked as *Cache Logic*, was developed and patented by Atmel.

Cache logic is conceptually similar to cache memory. In cache memory, the highest speed memory (usually SRAM) is used to store active data, while the bulk of data resides in lower cost storage, such as DRAM, EPROM, disk, etc. Cache logic works in a similar fashion. Only a small fraction of the circuitry is active in a system at any given time. Only active functions are loaded into the logic cache, while unused functions, or variations, reside in lower cost system memory. It is even possible to compile variations of a design in real time. Logic functions are loaded into the logic cache as required, replacing, or complementing the logic already present.

The ability to implement cache logic requires FPGAs that are capable of being dynamically reconfigured in system, either completely or partially, without disrupting the operation of the balance of logic in the device. Another requirement is architecture symmetry. This is necessary to enable the arbitrary placement of generic blocks in a location that is available as required. It is also necessary to allow for easy modeling of device characteristics for the artificial intelligence required in the partitioning of a design. The symmetry also simplifies the creation of arrays of devices to create a larger digital medium for the implementation of cache logic. Cache logic can be used in many applications. The example used in this description of cache logic will be a series of data acquisition products. The example will also discuss the concept of virtual products, which utilize the flexibility of programmable logic. Virtual products do not require cache logic programmability, but as we see the use of cache logic greatly reduces the amount of programmable digital media needed to implement a virtual product.

The Virtual Product

A *virtual product* is a combination of a "tangible asset," such as a data acquisition board and a service, such as product customization. The first thing to understand about virtual products is what the end customer wants, and how manufacturers can match theirs core competencies with these needs.

There are two issues raised in the manufacture of virtual products:

1. How to balance economy of scale achieved in volume manufacturing with special features that customers are willing to pay for; and
2. How to create diversity while maintaining a level of quality associated with standard high-volume production.

Cache Logic and FPGAs help the manufacturers achieve these two requirements of virtual products. A virtual product line is one that has characteristics that meet the needs of a class of customers. An example would be a PC based data acquisition product. Such a product has certain physical requirements consistent with a PC bus card standard. The board would also have a series of standard data gathering features, such as multiple channel A-to-D converters, digital I/O ports, D-to-A converters, and high speed clock counters. These features are implemented in standard, high integration ICs. The most complex portions of these products are the data path and protocol sections that connect the PC to the standard ICS products. The structure of this data path is a function of optimizing system performance, cost, and customer preference.

The traditional approach to creating data acquisition products is to create a board with a standard bus footprint, use industry standard A-to-D and D-to-A circuits, and then create a custom data path. The manufacturer then has to trust marketing studies and instinct to determine the best data path approach. It is possible to hedge the bet by adding redundancy. This redundancy has two detrimental effects: added cost, and added complexity for the end user. The selection of wrong data path protocol or excessive complexity caused by redundancy results in dissatisfied or nonexistent customers.

A virtual product does not mean that a manufacturer would be able to offer one product that was all things to all people. The use of programmable logic would allow a manufacturer create an extensive catalog of products, but only have a small number of tangible assemblies to tool for manufacturing. The manufacturer would use FPGAs and cache logic FPGAs to create diversity in their product line. The cost of diversity to the manufacturer is the cost of service, or "personalization engineering," required to create a niche design on a standard assembly. The advantage for the customer is a mass produced product that meets thier specific needs.

Quality

Quality is a very important advantage of the virtual product approach to design. Quality is usually the ultimate factor in device selection. Quality customer service can be defined as a high level of product diversity, and is measured by the degree that the specific customer requirements are met.

The foundation of product quality is in the manufacturing process. Unlike service quality, product diversity (quality customer service) is detrimental to the goal of 100% quality. Total product quality is attained by tooling for a long term process of implementation, evaluation, and feedback. Multiple custom products produced on the same assembly line conflict with the ability to attain the highest product quality and best economies of scale. Quality enhancing techniques, including just-in-time delivery, multi-discipline staffing, and integrated engineering, do not work well with multiple short run products.

The virtual product approach allows a manufacturer to perfect a single assembly. The FPGA's ability to be configured for self-test could even enhance quality of the assembly. Atmel has developed the IEEE 1149 boundary scan supermacro. Utilizing the reconfigurable logic capability of the AT6000 family, the boundary scan function may be loaded into the device, diagnostics performed, and then the device can be reconfigured for other logic functions. A single Atmel device may be used for testing and logic, with no overhead or speed penalty, as is the case for all other FPGAs and other ASIC devices.

The result would be an inventory of nearly identical raw product assemblies, which through virtual design becomes a catalog full of products when shipped to the customer. Most customer problems can be traced back to the virtual design personalization process, and be repaired in the field with FPGA configuration updates. It is also possible to introduce new features into virtual products as soon as they are conceived, rather than wait until a new hardware product is designed.

A New Paradigm in Customer and Supplier Relations

New ways of doing business will develop between suppliers and their customers in a "virtual product world." The customer must realize that the cost of product development is amortized in the customized virtual product. The cost of a new personalization may be nearly the same as a complete product.

A virtual product is obsolete when the need for a personalization ceases to exist, as opposed to when the product assembly wears out.

Data Acquisition Example

The Data Acquisition system shown in the accompanying figure is one where a family of products is desired but only one assembly will be manufactured. The intent is to offer a Multiple Channel Analog-to-Digital conversion and 16 digital I/O channels, and multiple digital timers for setting sample periods. The product will be a PC-AT bus card that supports various combinations of data transfer protocols. It is conceivable that different classes of customer will want use a specific setup exclusive of all others, and not understand why anyone would want to work any other way. Putting the features into a matrix shows how many potential products are possible from this set of components (see Table 1).

Table 1. ADC or Digital I/O Sample Mode

Bus Interface	External	Timer	Demand	State	Continuous
I/O Mapped	X		X		X
I/O with Internal	X	X		X	X
DMA Byte Transition	X	X	X	X	X
DMA Array Transition		X	X	X	X

The result is over 16 products that are different in their data path and protocol approach. It is possible that all of these protocols and data path structures can be implemented in a single 2,000- to 5,000-gate FPGA. Cache logic makes the fitting of the protocol in 2,000 to 5,000 gates possible. This is because each data path protocol combination still needs to be programmable. Items such as timer periods, I/O direction and grouping, the use of each interrupt and timer output, and DMA address counters can be implemented using cache logic, saving circuit redundancy and complexity.

Timer Periods

The timers can be compiled in a structured format. The user defines the number and period of timers required. Each timer contains an N-bit counter, and an M-value decoder, where M is the count equal to the desired period. The Atmel symmetric architecture makes the creation of macros from a high level description straight forward. An area for timers is included in the FPGA. The configuration data for that area is compiled according to customer description. The customer would receive a compiler program, and not a set of pre-defined configuration.

I/O Direction

The FPGA has flexible, individually programmable I/O. There is no reason that a description of the I/O direction and byte grouping could not result in the appropriate configuration of that area of the FPGA. The grouping of the I/Os into bytes is made by a cross-point switch. The cross-point switch is implemented using FPGA programmability. The only requirement in the development of the product is the allocation of sufficient resources in the FPGA for the cross-point area. A byte grouping description by the user would then result in configuration information for enabling the proper pass gate connections in that section of the FPGA.

Timer and Interrupt Usage

In a data acquisition circuit interrupts and timer outputs are used by many parts of the circuit. Time is required to set the period of a digital or analog sample. As the digital I/O control function, the interrupt and timer outputs can be routed to various combinations of control inputs, PC bus or internal logic of the data acquisition chips. This flexibility, although like a cross-point switch, would be more of a channel routing function. A set of routing resources is used for the connection of various programmable inputs and outputs with access to the reserved channel area.

DMA Address Generation

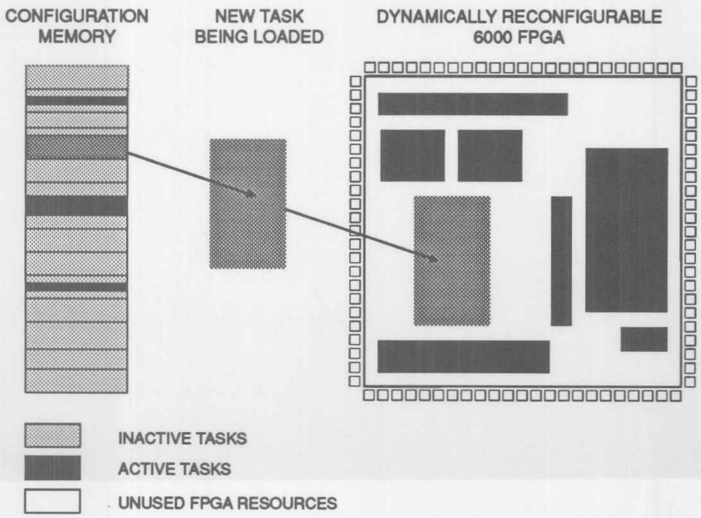
In designs using DMA, the DMA address is stored in a register or counter, depending on the mode. This counter is compiled based on software requirements.

FPGA Configuration

Figure 1 shows the configuration of the FPGA SRAM: controlling its functionality and hookup are a patchwork of configuration information from various sources. Some are a base configuration file used by all products, others from look-up tables or macro compilers. The Atmel FPGA would allow items such as timer periods, or DMA address values to be updated continuously, without effecting the operation of unrelated logic. The advantage of cache logic is the elimination of redundancy. Cache logic allows the creation of any structure from the digital medium. In traditional designs a circuit would have redundancy for each anticipated eventually, and the control to enable and disable it as well. Even if an application used all possible features, some circuits would be idle, since some combinations would never occur concurrently.

6

Figure 1. Cache Logic Concept



The creation of FPGA configuration files from a high-level description is not an easy task. The manufacturer of a virtual product will need to make the description of the final configuration as intuitive as possible for the end customer. Given equal raw product assemblies the virtual product, like the fixed product that is easiest to use, will be the most successful. The FPGA manufacturer's responsibility is to the manufacturer. The FPGA configuration and architecture must be easy to grasp, and the configuration process must be readily available. A set of compilation development tools and example like this data acquisition product will help to make the manufacturer more comfortable with the implementation of cache logic designs.

Benefits

There are several benefits of this approach to data-acquisition and other instrumentation suppliers:

1. New functionality may be added to existing hardware, without having to make modifications to the board.
2. The hardware may be tailored to the application, resulting in higher system performance across a broad range of applications.
3. Overall system reliability is improved by reducing the number of physical products and utilizing boundary scan macros for manufacturing and system testing.
4. Overall product life cycle costs are significantly reduced by using reusable software and hardware:
   - lower development costs
   - lower inventory costs
   - quicker time to market
   - fewer parts on the board
   - lower power consumption
   - lower total system cost

Summary

To many people the ideal of a virtual product is a futuristic concept. The FPGA of today is a virtual product. It is available and dependable. The Atmel FPGA and its abilities to implement cache logic make it a foundation for other virtual products. Design with this new technology has been commercially demonstrated. Today's design methodology, that requires a new product for each new function, will be replaced by the virtual product. The needs of both customers and suppliers, with customized products and improved quality; while reducing product development time and cost and attaining economies of scale by high volume production.

High-Speed, Loadable 16-Bit Binary Counter

Field Programmable Gate Array Application Note

6

By Frederick Furtek

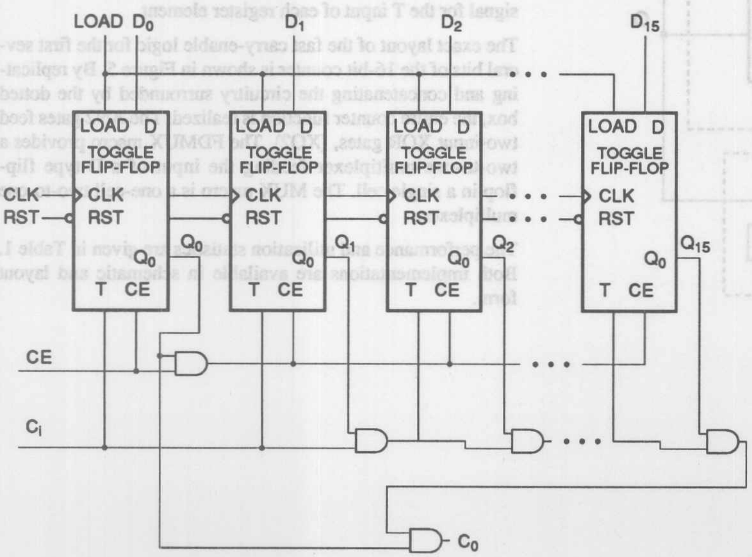
Introduction

The AT6000 Series field programmable gate array (FPGA) lets the designer implement a fast synchronous, loadable 16-bit binary counter that operates at 70 MHz on and off chip under the worst commercial operating conditions. The use of prescaled logic to generate the carry-enable signals for each count bit allows faster operation than traditional carry-enable generation methods. The 16-bit counter is very compact, yet the inputs and outputs are readily accessible.

Description

Figure 1 shows a block diagram representation of the counter architecture and I/O. CLK is the clock signal, RST is the reset signal, and LOAD is the load data signal. CE is the count enable signal. CLK is a positive, edge-triggered synchronous signal, RST is an active low, asynchronous signal, and LOAD is an active low, synchronous signal. Pins D_0 through D_{15} are the load data inputs, pins Q_0 through Q_{15} are the count bits. Pin C_i is the carry in, C_o is the carry out. Toggle flip-flops are used as the register elements for each bit of the counter.

Figure 1. Architecture and I/O of 16-Bit Counter



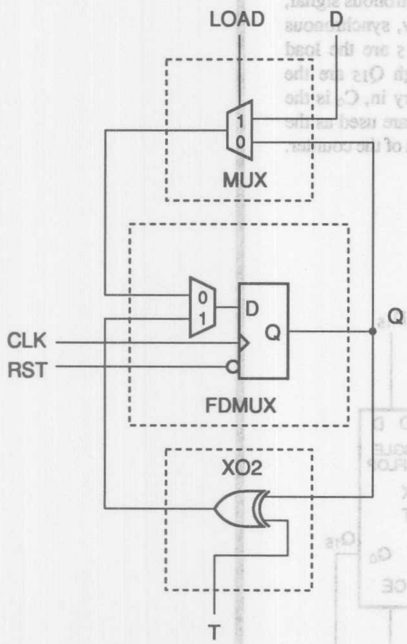
They toggle on the rising edge of CLK when their RST, CE, and T inputs pins are high and the LOAD pin is low.

Initial power-up of the AT6000 device resets all the registers so the counter begins counting on the first rising edge of CLK if C_i, RST, and CE are set high and LOAD is set low. The circuit counts by allowing each register element to toggle in succession on the rising edge of CLK if the Q outputs of all prior elements are asserted.

Asserting RST at any time inhibits counting, but also resets the registers to low values.

Figure 2 shows the implementation of a register element in the 16-bit counter logic architecture. The Q output of this circuit will toggle if T and CE are high and LOAD is low on the rising edge of CLK. If CE and LOAD are low, then Q will not change, regardless of T. If CE is high and LOAD is low, then Q will remain the same if T is low upon the rising edge of CLK.

Figure 2. Schematic of 16-Bit Counter Register Element



To load a value into the counter, LOAD is set high and CE is set low before the rising edge of CLK. The value is latched into the register elements on the rising edge of CLK.

CE should then be held low until after the next rising edge of CLK to allow the carry-enable logic time to recalculate the T inputs for each register element. The carry-enable logic is the chain of two-input AND gates that generates the T signal inputs.

If LOAD is asserted for one clock cycle and CE is low for two clock cycles, the data at $D_{0\text{-}15}$ is loaded into the registers on the first clock cycle, and the counting continues from the newly loaded value on the second cycle (Figure 3).

During the LOAD cycle, when the data at $D_{0\text{-}15}$ is clocked into the counter, the carry-enable logic must have time to generate and propagate the results to every bit. Since an arbitrary number at $D_{0\text{-}15}$ can cause a carry-enable signal to propagate along the entire length of the carry-enable chain, the critical path during a LOAD operation has the potential to pass through 14 AND gate (AN2) stages before entering the last register element. By holding the CE signal low an extra clock cycle to inhibit the counting operation (as shown in Figure 3), the carry-enable logic has additional time to propagate the correct values to each bit.

During normal operation Q_0, the least-significant bit of the counter, is also a fast carry-enable signal. As shown in Figure 4, the CE inputs of each register element ahead of the first bit are tied to the Q_0. All bits greater than Q_0 must wait for Q_0 to switch from low to high before they can change on the rising edge of CLK. As the more significant bits change, their values trickle forward through the two-input AND gates that form the carry-enable logic to the T inputs of succeeding register elements. Distributing Q_0 in this manner allows an extra clock cycle for the chain of two-input AND gates to calculate the carry-enable signal for the T input of each register element.

The exact layout of the fast carry-enable logic for the first several bits of the 16-bit counter is shown in Figure 5. By replicating and concatenating the circuitry surrounded by the dotted box, the entire counter function is realized. The AN2 gates feed two-input XOR gates, (XO2). The FDMUX macro provides a two-to-one multiplexer feeding the input of a D-type flip-flop in a single cell. The MUX macro is a one-cell two-to-one multiplexer.

The performance and utilization statistics are given in Table 1. Both implementations are available in schematic and layout form.

Figure 3. Timing Diagram of Counter Load Cycle

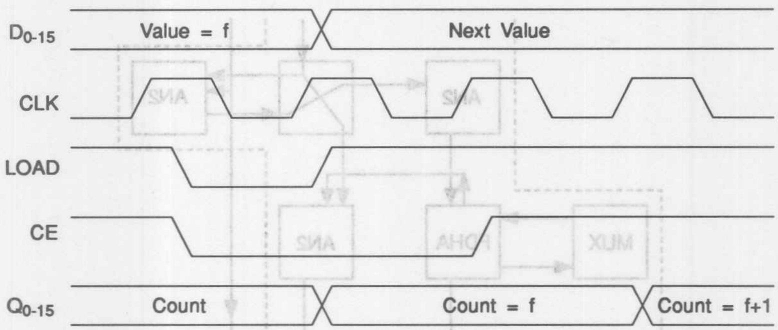


Figure 4. Schematic of Counter Architecture

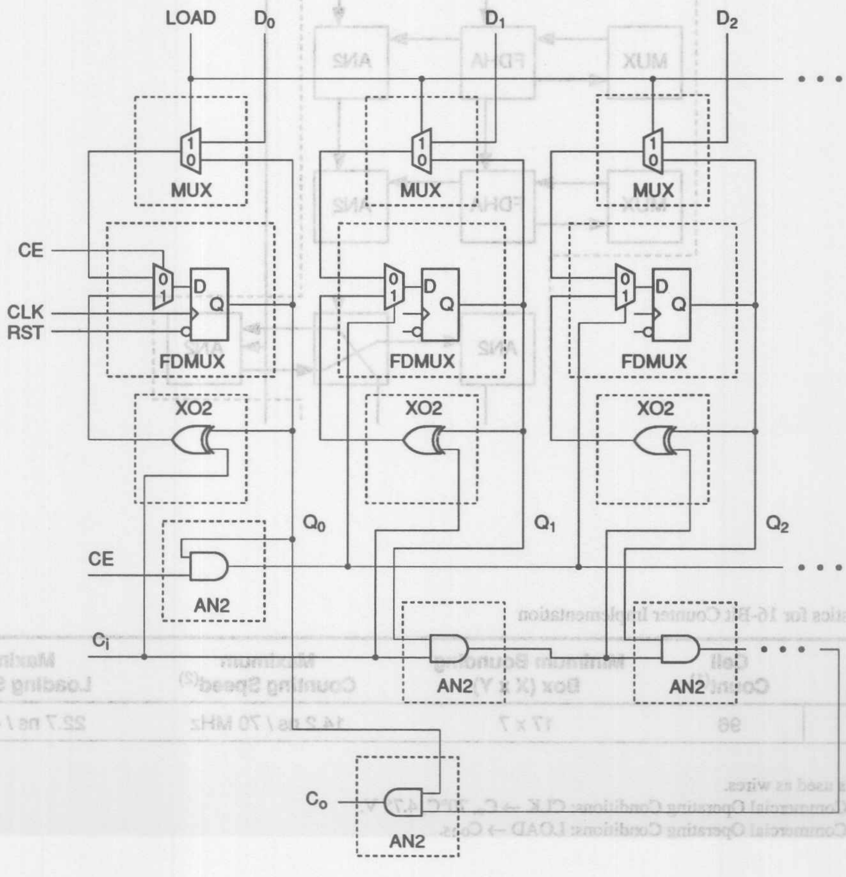


Figure 5. Layout of Counter Architecture

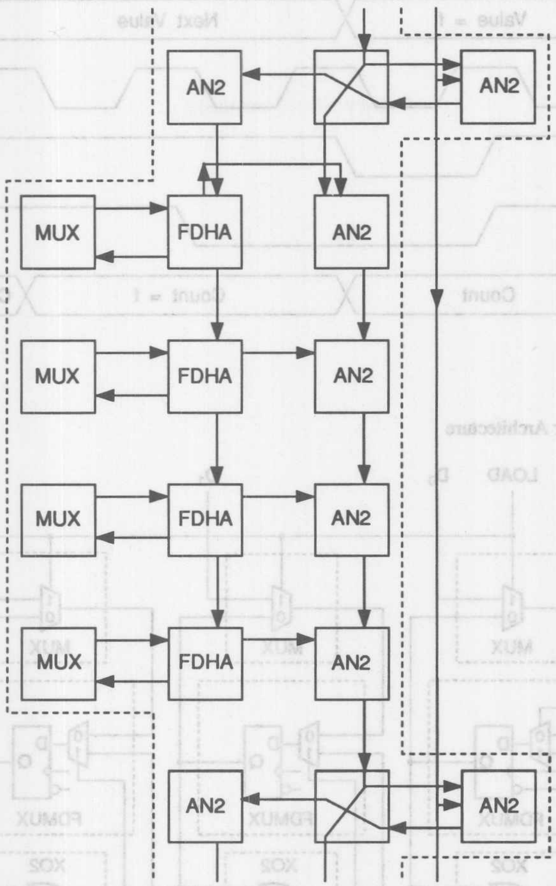


Table 1. Statistics for 16-Bit Counter Implementation

Counter	Cell Count[1]	Minimum Bounding Box (X x Y)	Maximum Counting Speed[2]	Maximum Loading Speed[3]
16-Bit	96	17 x 7	14.2 ns / 70 MHz	22.7 ns / 44 MHz

Notes:
1. Includes cells used as wires.
2. Worst-Case Commercial Operating Conditions: CLK → C_o, 70°C, 4.75 V.
3. Worst-Case Commercial Operating Conditions: LOAD → $C_{0\text{-}15}$.

Compact, Loadable 16- and 32-Bit Binary Counters

Field Programmable Gate Array

Application Note

Introduction

The AT6000 Series architecture accommodates dense, synchronous, loadable binary counters. A 16-bit counter counts at 42 MHz, and a 32-bit at 36 MHz in AT6000-2 devices. Both counters are very compact, yet their inputs and outputs are readily accessible.

Description

Figure 1 is a block diagram representation of the I/O and architecture for a 16- or 32-bit counter. Pin CLK is the clock signal, RST the reset signal, and LOAD the load data signal. CLK is a positive, edge-triggered synchronous signal, and LOAD is an active low, synchronous signal. Pins D_0 through $D_{15,31}$ are the load data inputs, and pins Q_0 through $Q_{15,31}$ are the count bits. Pin C_i is the carry in; C_o is the carry out.

4-bit synchronous loadable binary counters are used to compose larger 16- or 32-bit counters. These 4-bit counters toggle on the rising edge of CLK when their RST is high and LOAD is low.

Initial power-up of the AT6000 device resets all the registers. The counter begins counting on the first rising edge of CLK if C_i and RST are set high and LOAD is set low. The circuit counts by allowing each 4-bit counter stage to toggle in succession on the rising edge of CLK if the Q outputs of all prior stages are asserted. Asserting RST at any time inhibits counting, but also resets the 4-bit counters to low values.

Figure 1. Architecture and I/O of 16-Bit or 32-Bit Counter

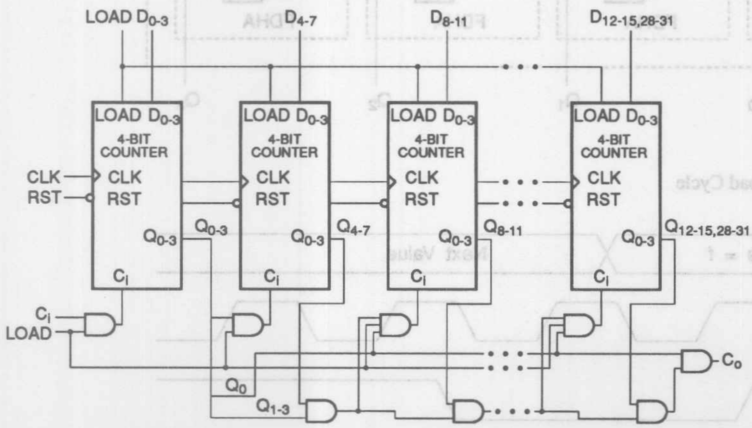


Figure 2 shows the implementation of the 4-bit counter stage used in the 16- and 32-bit counter architectures. A Q output in this circuit will toggle if C_i is high and LOAD is low, and all prior Q output are asserted upon the rising edge of CLK. If C_i and LOAD are low, then Q will not change. This circuit exists as a predefined marco library element called CRP4.

To load a value into a 16-bit counter, LOAD is set high prior to the rising edge of CLK. The value is latched into the CRP4 stages on the rising edge of CLK.

LOAD should then be held low until after the next rising edge of CLK to allow the carry-enable logic time to recalculate the carry-enable bit for each CRP4 stage. The carry-enable logic is the chain of two-input AND gates that generates the C_i signal inputs for each CRP4 stage. In Figure 3, if LOAD is asserted for two clock cycles the data at $D_{0\text{-}15}$ is loaded into the CRP4 macros on the first clock cycle, and counting continues from the newly loaded value two cycles later

During the LOAD cycle, when the data at $D_{0\text{-}15}$ is clocked into the 16-bit counter, the carry-enable logic must have time to generate and propagate the results to every bit. Since an arbitrary number at $D_{0\text{-}15}$ can cause a carry-enable signal to propagate along the entire length of the carry-enable chain, the critical path during a LOAD operation has the potential to pass through 17 AND gates (AN2) before entering the last register element. By holding the LOAD signal low an extra clock cycle to inhibit the counting operation (as shown in Figure 3), the carry-enable logic will have additional time to propagate the correct values to each bit.

Figure 2. Schematic of a 4-Bit Counter Macro CRP4

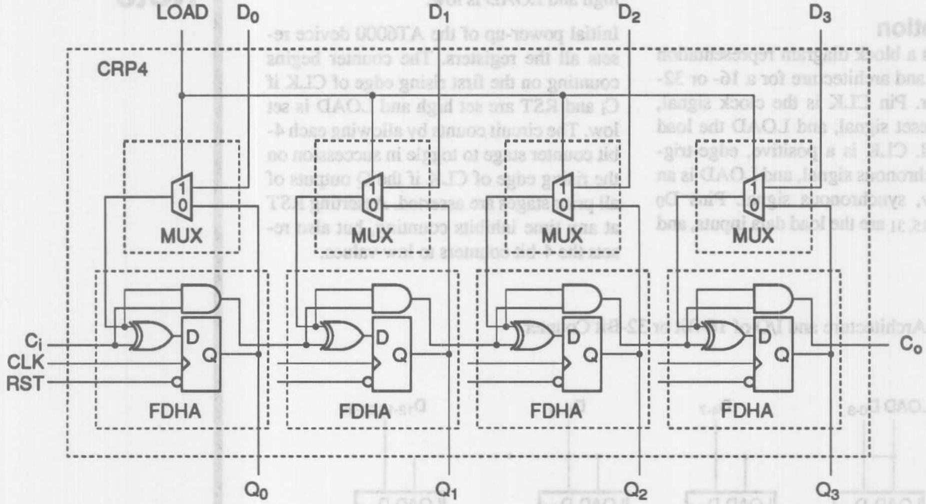


Figure 3. Timing Diagram of Counter Load Cycle

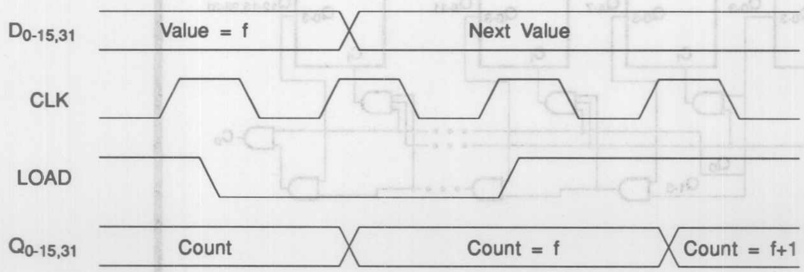


Figure 4 shows the detailed schematic of a 16-bit counter partitioned into four stages. During normal operation Q_0, the least-significant bit of the counter, is also a fast carry-enable signal. All bits greater than Q_0 must wait for Q_0 to switch logic levels before their carry-enable logic stabilizes. Q_0 is distributed to all four CRP4 macros in an attempt to balance and minimize the propagation delay of Q_0 to the C_i of the more significant CRP4 macros. For example, as the more significant bits in the first stage are asserted, their values trickle through the two-input AND gates (AN2) that form part of the carry-enable logic. When all the more significant bits are asserted and Q_0 switches from low to high on the rising edge of CLK, the carry-enable signal to the C_i input of the second CRP4 is enabled. While its C_i signal is asserted, the CRP4 in the second stage counts on the rising edge of CLK.

By replicating and concatenating the circuitry surrounded by the dotted box, larger counter functions are realized. The performance and utilization statistics for the 16- and 32-bit counters are given in Table 1. Both implementations are available in schematic and layout form.

Figure 4. Schematic of Counter Architecture

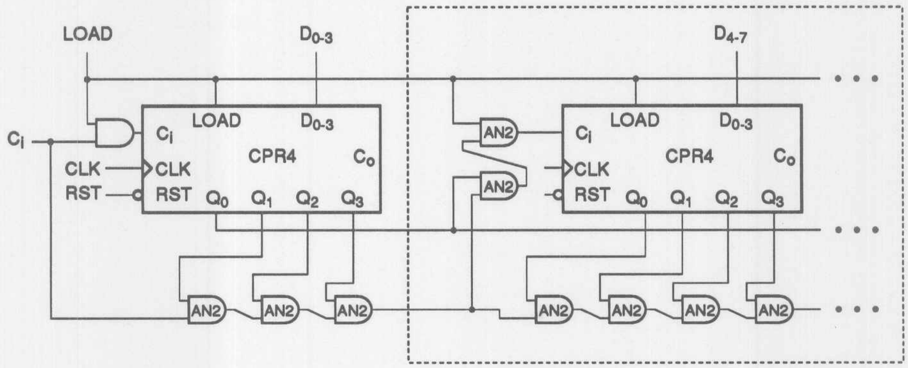


Table 1. 16- and 32-Bit Counter Performance Comparison

Counter	Cell Count[1]	Minimum Bounding Box (X x Y)	Maximum Counting Speed[2]	Maximum Loading Speed[3]
16-Bit	76	10 x 8	22 ns / 45 MHz	55.5 ns / 18 MHz
32-Bit	136	18 x 8	35.7 ns / 28 Mhz	83.3 ns / 12 MHz

Notes:
1. Includes cells used as wires.
2. Worst-Case Commercial Operating Conditions: CLK $\rightarrow C_o$, 70°C, 4.75 V.
3. Worst-Case Commercial Operating Conditions: LOAD $\rightarrow Q_{0\text{-}15,\,31}$.

16-Bit Up/Down Counter/Shift Register

Field Programmable Gate Array

Application Note

Introduction

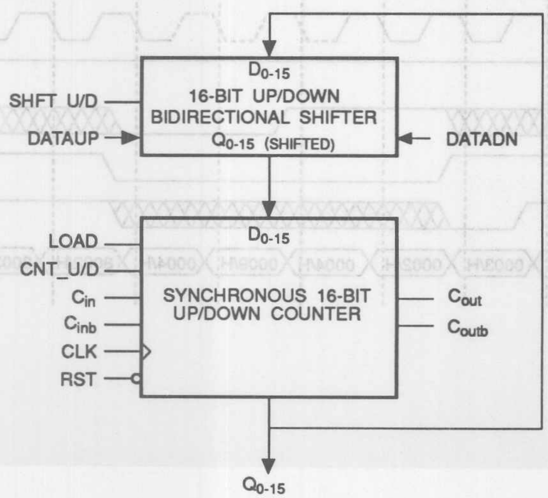
The AT6000 Series field programmable gate array (FPGA) lets the designer implement a synchronous, 16-bit Up/Down Counter/Shift Register that operates at 22 MHz under the worst commercial operating conditions. In this circuit is most of the combined functionality of the 74193 Up/Down Counter and 74194 Bidirectional Shift Register TTL components. It would take eight discrete TTL components to implement an 16-bit Up/Down Counter/Shift Register. Nearly the same function can be achieved in a AT6005 using less than 8% of the available logic.

Description

Figure 1 shows a block diagram of the counter/shifter. Pin CLK is the clock signal, RST the reset signal, and LOAD the count/shift control signal. CLK is a positive, edge-triggered synchronous signal, RST an active low, asynchronous signal, and LOAD an active low, synchronous signal. Pins Q_0 through Q_{15} are the count bits. Pins C_{in} and C_{out} are the carry-in and carry-out signals. Pins C_{inb} and C_{outb} are the carry-in borrow and carry-out borrow signals. SHFT_U/D and CNT_U/D control the direction of the shift and count operations. DATAUP and DATADN are the shift-up and shift-down serial data inputs for the shifting operation.

The output of a 16-bit fast ripple-carry counter is input into a bidirectional shifter that shifts the data inputs up or down one bit position. The output of the shifter is fed back to the parallel data inputs (D_{0-15}) of the counter. When LOAD is set high, the output of the shifter is ignored and the counter increments or decrements depending on the CNT_U/D. If LOAD becomes unasserted, the counting operation is inhibited, and the shifter provides the shifted count at Q_{0-15} to the parallel data inputs of

Figure 1. 16-Bit Up/Down Counter/Shifter



the counter. While LOAD remains low, the counter essentially acts as a shift register, and will shift the data at Q_{0-15} either up or down depending on SHFT_U/D.

Initial power-up of the AT6000 device resets all the registers in the counter/shifter. While LOAD and RST are asserted, incremental counting commences if CNT_U/D and C_{in} are asserted before the rising edge of CLK. Decrement counting commences if CNT_U/D is unasserted and C_{inb} is asserted before the rising edge of CLK.

If LOAD is set low, the circuit becomes a shift register on the rising edge of CLK. The CNT_U/D, C_{in}, and C_{inb} signals are ignored, and control of the circuit is determined by SHFT_U/D, DATAUP, and DATADN. If SHFT_U/D is asserted, the data at Q_n will be shifted to Q_{n+1} on the rising edge of CLK. The value present at DATAUP is shifted to Q_0. Down shifting occurs when SHFT_U/D is set low.

The Up/Down Counter/Shifter must be serially loaded with a starting value.

Although Figure 1 shows that the 16-bit counter has a parallel load capability, the data inputs of the counter are already driven by the shifter. While LOAD is low, counting is inhibited. With the proper control of SHFT_U/D, any 16-bit value can be shifted serially into the register through the inputs DATAUP or DATADN. When the register is loaded with the correct value, Up/Down counting can begin on the first rising edge of CLK after LOAD is set high. Parallel loading of the start value is the only feature not inherent in the circuit that is present in the 74193 TTL device. Figure 2 shows the timing of the count/shift operation.

The schematic in Figure 3 shows the detailed implementation of the circuit. The FDHA macro is a half-adder sum that feeds a D-type flip-flop. Together with the MUX two-to-one multiplexer macros and the SELBUFS selector macros, an Up/Down counter with parallel load is constructed using only six cells per bit. SELBUFS macros enable the carry generation logic for each bit of the counter to be implemented in only one cell. The chain of SELBUFS macros that forms the carry-generation logic is also the critical path of the circuit. In the layout of this counter the least-significant bit Q_0 is distributed to the carry-enable logic of the other more significant bits. Distributing Q_0 will improve the performance by minimizing the delay through the critical path for both up-counting and down-counting.

MUX macros are used to form the shifter portion of the circuit. The inputs into each MUX of the shifter will be the Q_{n-1} and Q_{n+1} outputs of the counter, and the output of the shifter will feed the n^{th} parallel data input of the counter. At the least- and most-significant bits, the MUX macros will use DATAUP and DATADN as inputs.

The performance and utilization statistics for the 16-bit Up/Down Counter/Shifter are given in Table 1. This implementation is available in schematic and layout form.

Figure 2. Timing of Load/Count Operation

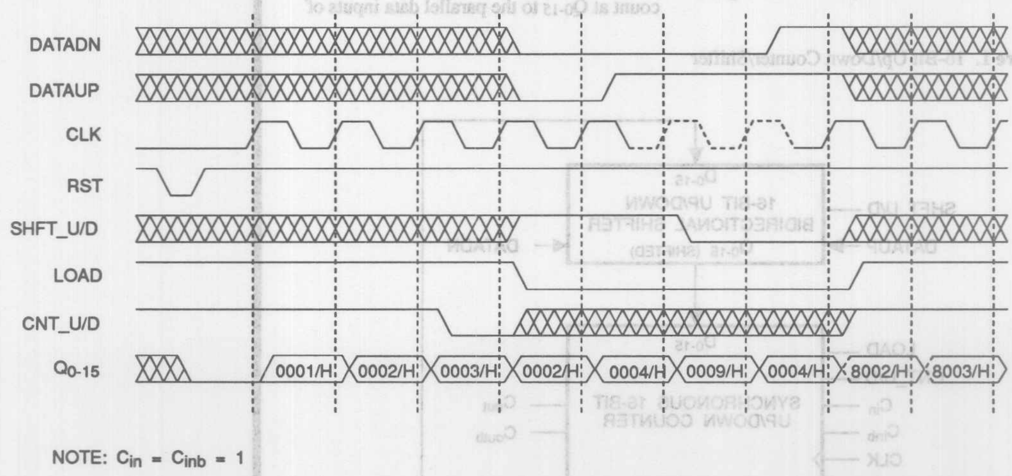


Figure 3. Schematic of Counter Figure

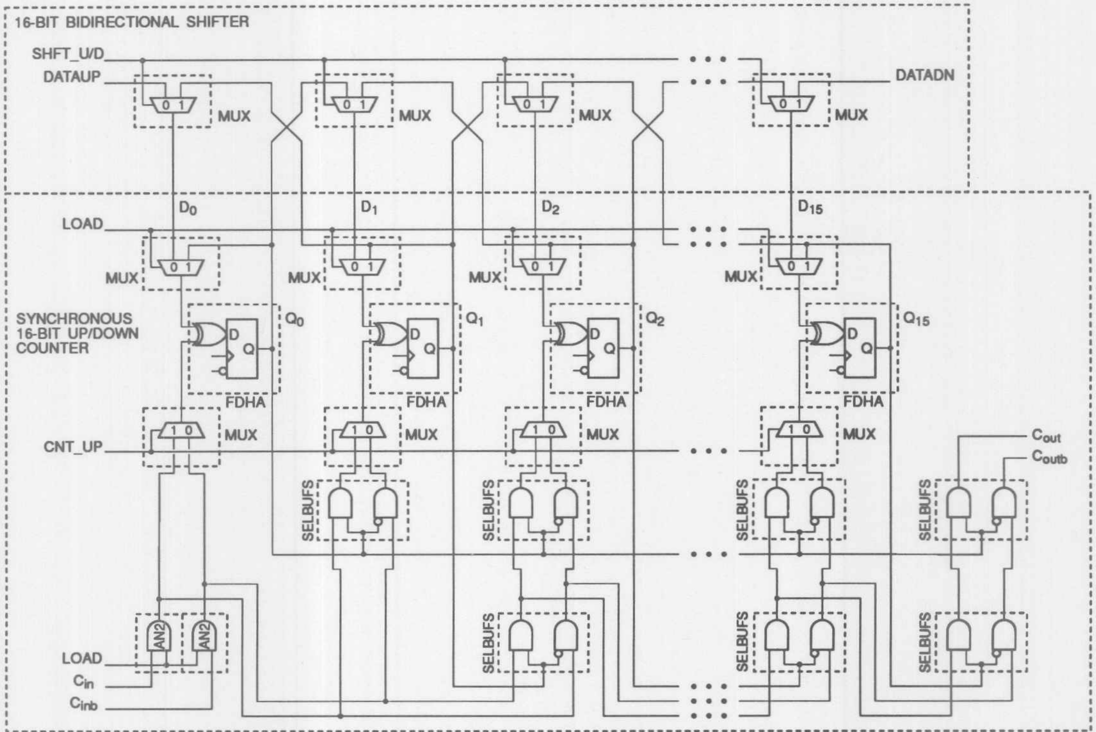


Table 1. Statistics 16-Bit Up/Down Counter/Shift Register

Counter/Shift Register	Cell Count(1)	Minimum Bounding Box (X x Y)	Maximum Speed(2)
16-Bit	215	13 x 18	45.5 ns / 22 MHz

Notes:
1. Includes cells used as wires.
2. CLK → $C_{out, outb}$. Worst-Case Commercial Operating Conditions: 70°C, 4.75 V.

9-Bit Programmable Terminal Counter

Field Programmable Gate Array

Application Note

Introduction

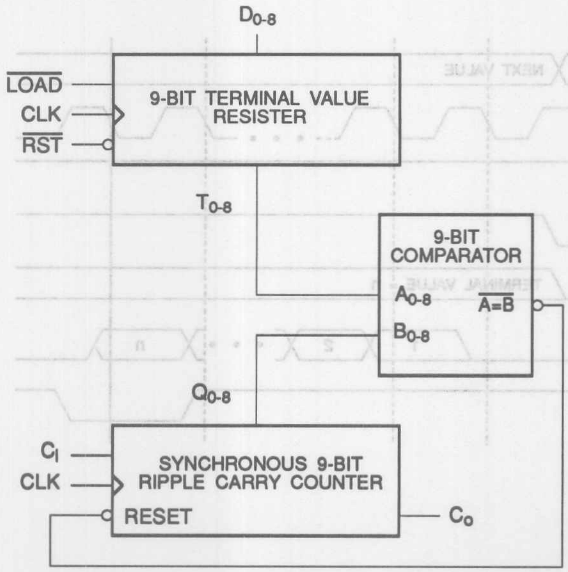
The AT6000 Series field programmable gate array (FPGA) lets the designer implement synchronous, programmable 9-bit terminal counters optimized for speed or layout area. A high-performance version is available that can operate at 33 MHz under the worst commercial operating conditions. If layout area is a consideration, a 33% smaller version is available that can still operate at 28 MHz worst case. An additional feature inherent in both counters is the ability to continue counting while a terminal value is being loaded into the terminal register.

Description

Figure 1 shows a block diagram of the counter. Both versions have essentially the same architectural structure. Pin CLK is the clock signal, RST the reset signal, and LOAD the load terminal value signal. CLK is a positive, edge-triggered synchronous signal, RST an active low, asynchronous signal, and LOAD an active low, synchronous signal. Pins D_0 through D_8 are the terminal value inputs, and pins Q_0 through Q_8 are the count bits. Pins C_i and C_o are the carry-in and carry-out signals.

Initial power-up of the AT6000 device resets all the registers in the counter and terminal register. Counting does not com-

Figure 1. 9-Bit Programmable Terminal Counter



through the inputs D_{0-8} by holding LOAD low during the rising edge of CLK. Assuming C_i is asserted, the counter increments on the rising edge of CLK until the terminal value is reached on the outputs Q_{0-8}. On the next rising edge of CLK, the counter is synchronously reset to zero. Figure 2 shows the timing of the terminal value load and count operations for the circuit.

The schematic in Figure 3 shows the detailed implementation of the high-speed version of the circuit. The FDMUX macro is a two-to-one multiplexer that feeds a D-type flip-flop. A terminal value register composed of FDMUX macros lets a terminal value be loaded synchronously through D_{0-8}. The terminal value register holds the previously loaded value at its outputs. Counting does not commence until a non-zero value is loaded into the terminal value register.

A synchronous ripple-carry counter begins counting after a terminal value has been loaded into the terminal register. The counter is composed of FD, XO2, and AN2 macros—D-type flip-flops, two-input exclusive OR gates, and two-input AND gates respectively. Outputs Q_{0-8} feed into a comparator circuit composed of INV inverter gate, XO2, and AN2 gates. The output of the comparator controls the initialization of the counter by checking Q_{0-8} against the terminal value. When Q_{0-8} is

The critical path of the circuit starts at Q_0, the first bit of the counter, travels through the comparator, and then back to the input of the first counter bit. Performance is enhanced by breaking the chain of AN2 gates in the comparator into three- and four-gate segments (Figure 4), then combining the output of each segment with a two-input NAND macro called ND2. By gating the D-type flip-flops of the counter with the output of the comparator, path delay is minimized and performance is enhanced.

Figure 5 shows the detailed schematic for the compact version. The terminal value register is exactly the same as in the high-speed version, but the outputs of the register are not inverted as they enter the 9-bit comparator. Instead, the outputs of the counter are inverted as they enter the comparator, thus improving layout compactness. An FDXOAN3 macro, a complex gate structure feeding a D-type flip-flop, is used to implement the 9-bit synchronous ripple-carry counter in only nine cells. As in the high-speed version, the critical path also resides in the comparator portion of the circuit.

The performance and utilization statistics for both versions are given in Table 1. Both implementations are available in schematic and layout form.

Figure 2. Timing of Load/Count Operation

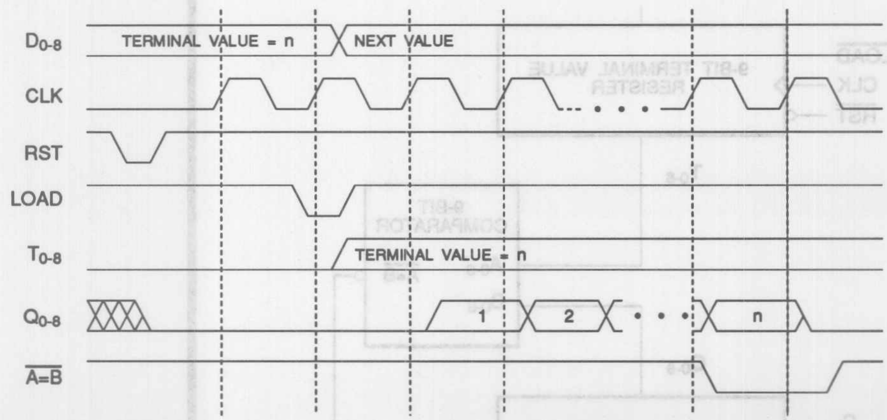


Figure 3. High-Speed Programmable Terminal Counter Architecture

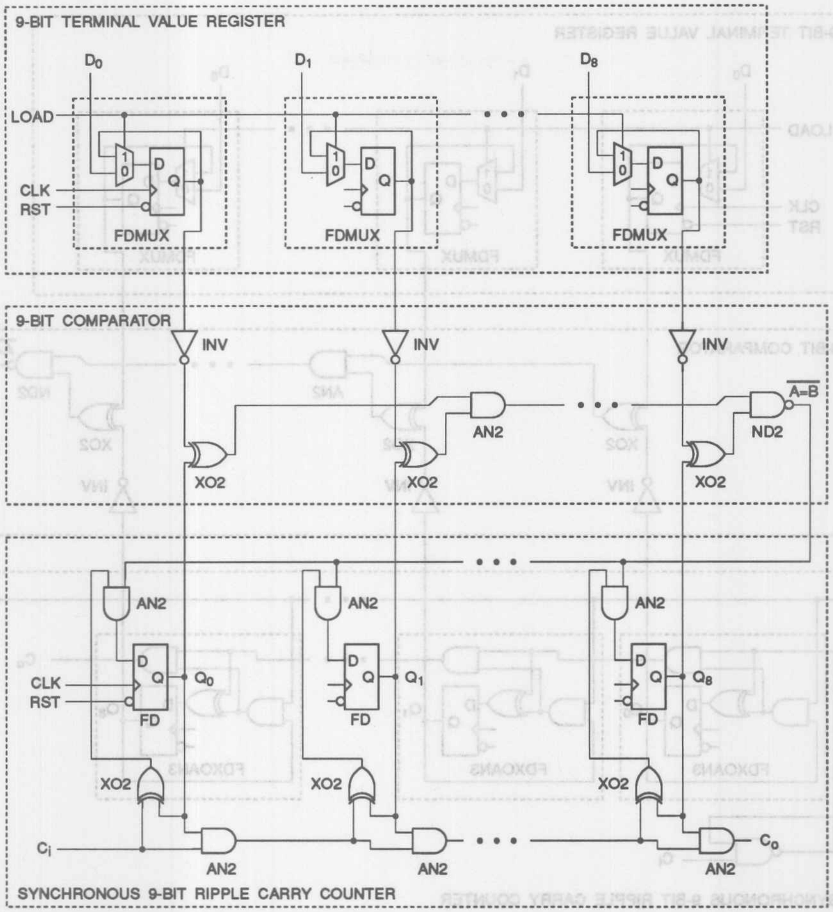


Figure 4. Schematic of Comparator Circuit

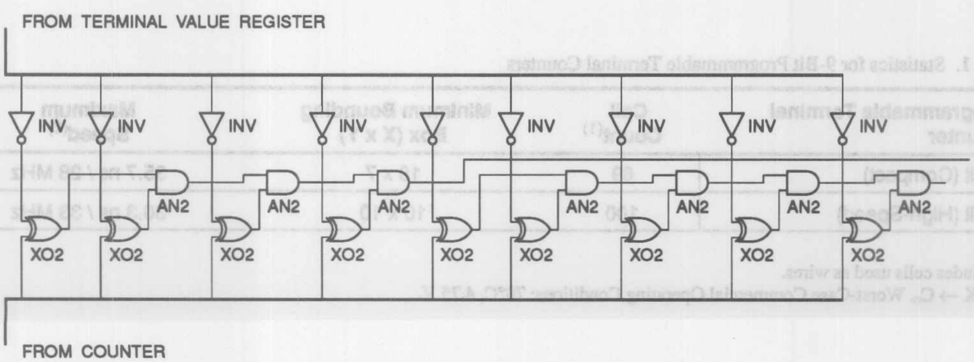


Figure 5. Compact Programmable Terminal Counter Architecture

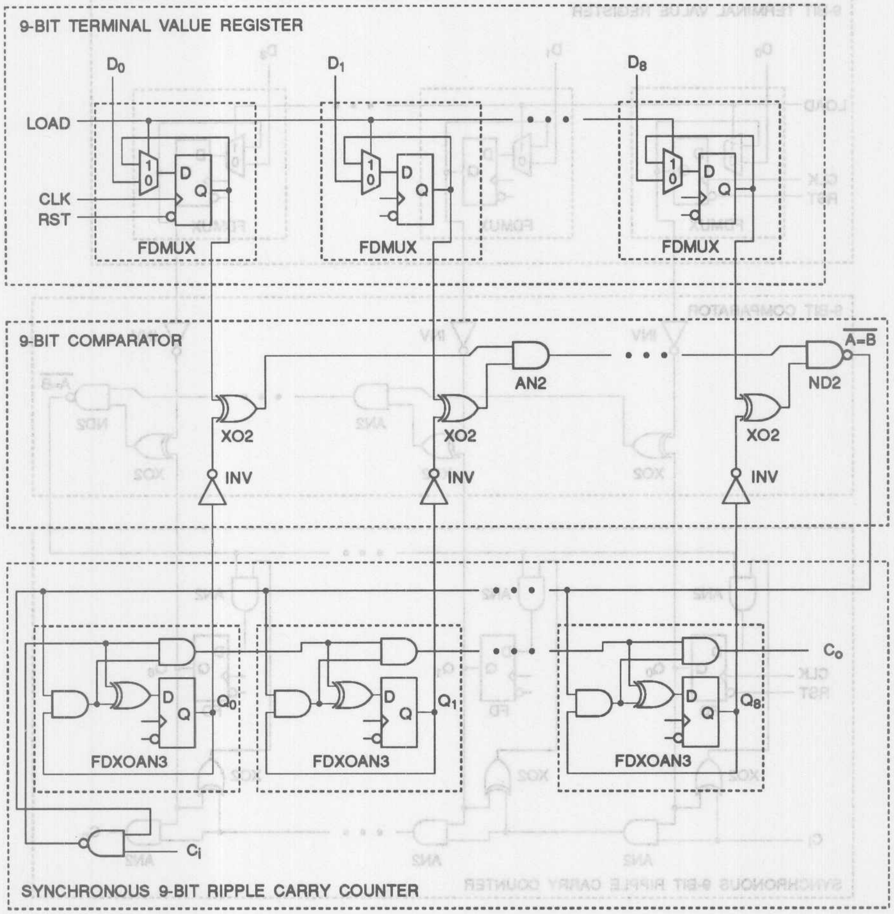


Table 1. Statistics for 9-Bit Programmable Terminal Counters

Programmable Terminal Counter	Cell Count(1)	Minimum Bounding Box (X x Y)	Maximum Speed(2)
9-Bit (Compact)	69	10 x 7	35.7 ns / 28 MHz
9-Bit (High-Speed)	100	10 x 10	30.3 ns / 33 MHz

Notes:
1. Includes cells used as wires.
2. CLK → C_o. Worst-Case Commercial Operating Conditions: 70°C, 4.75 V.

16-Bit Carry-Select Adder

By Frederick Furtek

Introduction

Ripple-carry adders are the simplest and most compact adders (they require as little as four cells per bit in the AT6000 architecture), but their performance is limited by a carry that must ripple from the least-significant to the most-significant bit. A carry-select adder implemented in the AT6000 achieves speeds 40% to 90% faster by performing additions in parallel and reducing the maximum carry path.

Description

A carry-select adder is divided into sectors, each of which—except for the least-significant—performs two additions in parallel, one assuming a carry-in of zero, the other a carry-in of one. The 16-bit carry-select adder of Figure 1, for example, is divided into sectors of lengths 1, 2, 3, 4, and 6, proceeding from least-significant to most-significant bit. The 4-bit sector of Figure 2 illustrates the general principle.

Within the sector, there are two 4-bit ripple-carry adders receiving the same data inputs but different carry-ins. The upper adder has a carry-in of zero; the lower adder a carry-in of one. The actual carry-in from the preceding sector selects one of the two adders. If the carry-in is zero, the sum and carry-out of the upper adder are selected. If the carry-in is one, the sum and carry-out of the lower adder are selected.

Logically, the result is no different than if a single ripple-carry adder were used. The difference, of course, is in performance. Instead of having to ripple through four full adders, the carry now only has to pass through a single multiplexer. In the AT6000 implementation (Figure 3), that multiplexer is implemented in a single cell, and the carry path through the sector incurs only a wire delay, a local-bus delay, and a multiplexer delay. Table 1 lists sizes and speeds for 16-bit ripple-carry and carry-select adders implemented in the AT6000.

Figure 1. 16-Bit Carry-Select Adder

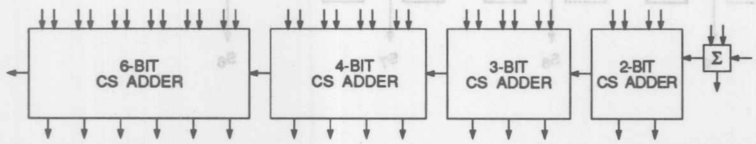


Figure 2. 4-Bit Sector (Schematic)

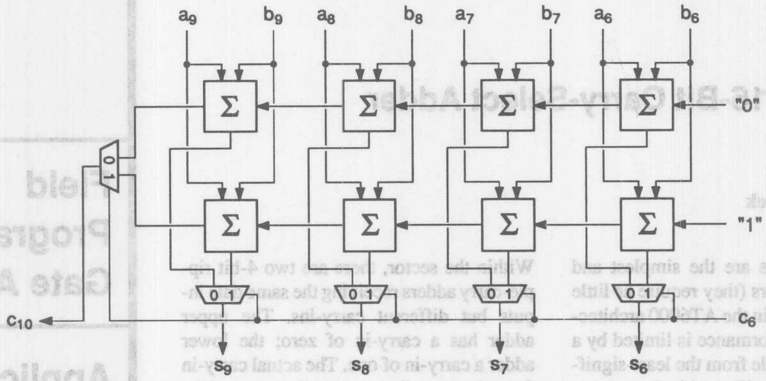


Figure 3. 4-Bit Sector (Layout)

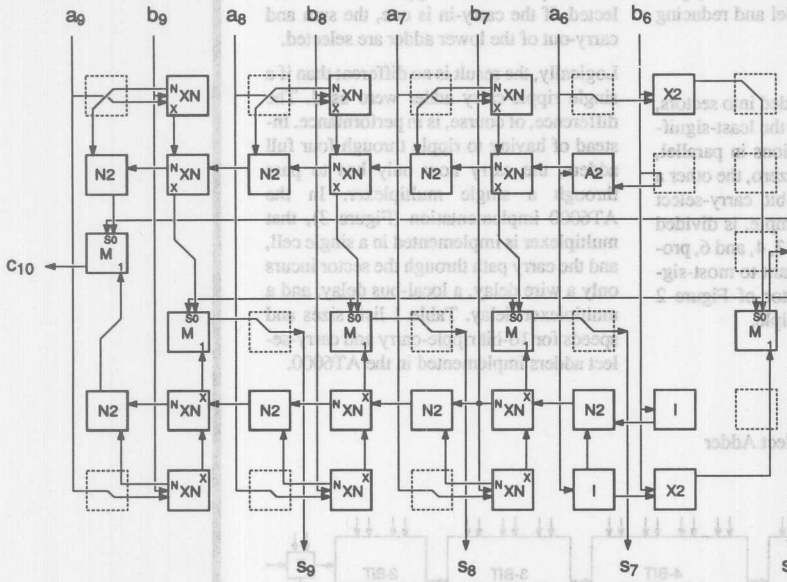


Table 1. Statistics for 16-Bit Adders

16-Bit Adder	Cell Count(1)	Minimum Bounding Box (X x Y)	Maximum Speed (-4)(2)	Maximum Speed (-2)(2)
Ripple Carry	64	2 x 32	111.9 ns / 8.9 MHz	67.7 ns / 14.7 MHz
Fast Ripple Carry	96	6 x 16	87.2 ns / 11.4 MHz	51.6 ns / 19.3 MHz
Carry Select	222	6 x 37	63.4 ns / 15.7 MHz	35.8 ns / 27.9 MHz

Notes:
1. Includes cells used as wires.
2. Worst-Case Commercial Operating Conditions: 70°C, 4.75 V.

Field Programmable Gate Array

Application Note

Ripple-Carry Adders

By Frederick Furtek

Introduction

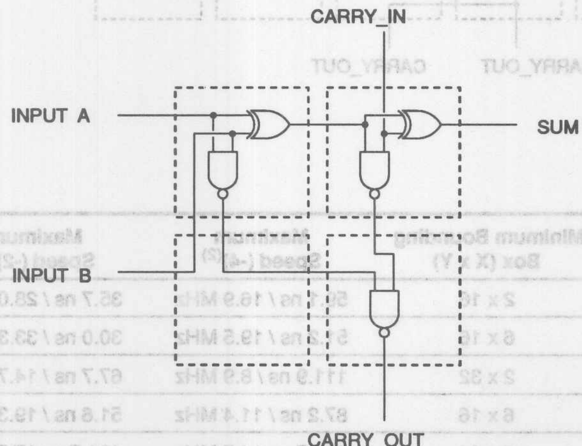
With a NAND and an XOR available simultaneously in a single cell, the AT6000 architecture is ideally suited for implementing arithmetic operations, including parallel adders. Ripple-carry adders—the simplest and most compact parallel adders—require as little as four cells per bit, and one layout has a carry delay of only one cell per bit.

Description

In the AT6000 architecture, a NAND and an XOR—basic building blocks of binary arithmetic—are available simultaneously in a single cell. The NAND/XOR is used in making full adders (FAs), which, in turn, are used in creating parallel adders.

A full adder has three binary inputs—two addends and a carry_in, and two outputs—sum and carry_out. The sum is the exclusive OR (XOR) of the three inputs, while carry_out is the majority (two out of three) of the three inputs. The simplest and most compact full-adder layout in the AT6000 architecture uses just four cells (Figure 1). The carry_in and carry_out, moreover, are aligned so that an *n*-bit adder occupying *4n* cells is created by simply abutting *n* full adders. An 8-bit parallel adder constructed from these adders uses only 32 cells (Figure 2).

Figure 1. Full Adder: Compact Layout



6

A second full-adder layout (Figure 3) uses six cells, plus a local bus, but the carry now propagates through only one cell per bit instead of the two cells per bit of the adder in Figure 1. This reduced delay in the carry path produces ripple-carry adders that run about one-third faster. An examination of the circuit shows that the sum output is still the XOR of the three full-adder inputs (the adder has only three distinct inputs and two distinct outputs; the carry_in and carry_out signals are replicated to satisfy the needs of the layout). The carry_out is still the majority of the three inputs although it is now constructed from two AND gates feeding an XOR. A little Boolean algebra shows that the function is identical to the three NAND gates used above (Figure 1) to produce the carry_out.

The size and performance of various ripple-carry adders are summarized below for the -4 and -2 speed grades (Table 1).

Figure 2. 8-Bit Ripple-Carry Adder

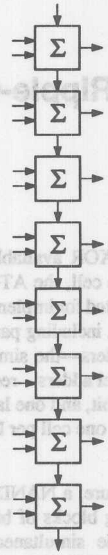


Figure 3. Full Adder: Fast Layout

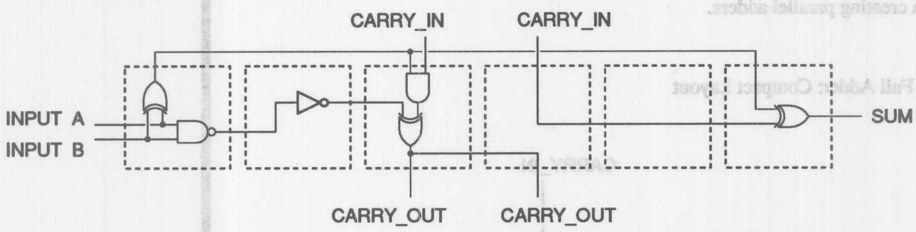


Table 1. Statistics for Ripple-Carry Adders

Adder	Cell Count)[1]	Minimum Bounding Box (X x Y)	Maximum Speed (-4)[2]	Maximum Speed (-2)[2]
8-Bit Ripple Carry	64	2 x 16	59.1 ns / 16.9 MHz	35.7 ns / 28.0 MHz
8-Bit Fast Ripple Carry	48	6 x 16	51.2 ns / 19.5 MHz	30.0 ns / 33.3 MHz
16-Bit Ripple Carry	64	2 x 32	111.9 ns / 8.9 MHz	67.7 ns / 14.7 MHz
16-Bit Fast Ripple Carry	96	6 x 16	87.2 ns / 11.4 MHz	51.6 ns / 19.3 MHz
32-Bit Ripple Carry	128	2 x 64	217.5 ns / 4.5 MHz	131.7 ns / 7.5 MHz
32-Bit Fast Ripple Carry	192	6 x 32	159.2 ns / 6.2 MHz	94.8 ns / 10.5 MHz

Notes:
1. Includes cells used as wires.
2. Worst-Case Commercial Operating Conditions: 70°C, 4.75 V.

Barrel Shifter

Field Programmable Gate Array

Application Note

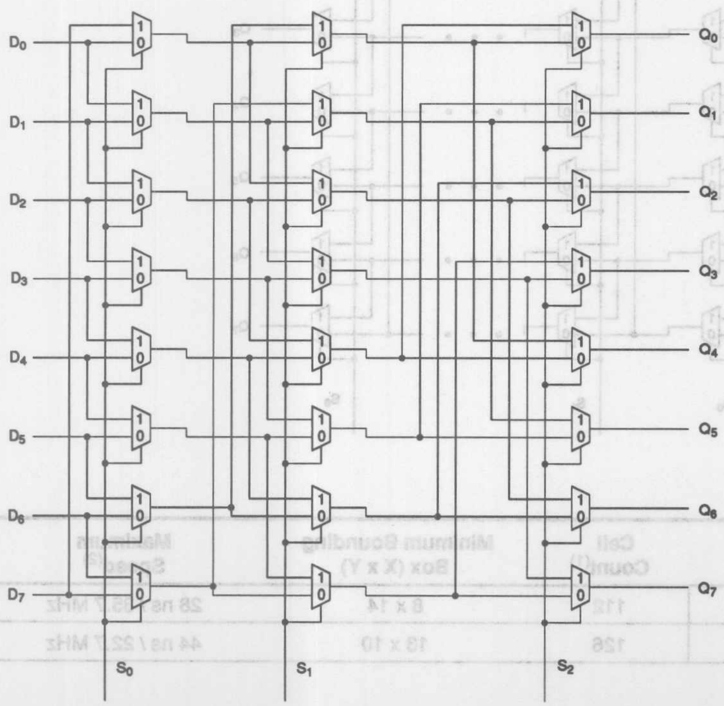
Introduction

The AT6000 Series field programmable gate array (FPGA) allows the designer to implement fast compact 8-bit barrel shifters, and modular shifters that can be easily sized for specific needs. Performance is enhanced by a unique feature of the busing architecture that enables the select control lines to be distributed across the data path with minimal skew, and a cell architecture that allows a two-to-one multiplexer (MUX) to be realized in just one cell.

Description

Figure 1 shows the fast, compact barrel shifter. Depending on the encoded shift control lines, $S_{0\text{-}2}$, the data inputs $D_{0\text{-}7}$ are shifted when they reach the outputs $Q_{0\text{-}7}$. If S_0 is asserted and S_1 and S_2 are unasserted, the value at $D_{0\text{-}7}$ is passed to the next most significant Q output. For example, D_0 is passed to Q_1, and D_1 passed to Q_2, while D_7 wraps around and is passed to Q_0. If none of the shift controls are asserted, the data inputs $D_{0\text{-}7}$ are passed to the corresponding outputs $Q_{0\text{-}7}$ without being shifted.

Figure 1. Fast, Compact Barrel Shifter with Encoded Shift Control



coded such that the data is shifted one bit position for every shift line that is asserted. The bit-width and depth of the modular barrel shifter can be easily expanded because of the efficient interconnections network. Each multiplexer output connects to its two nearest neighbors in the next column. The multiplexers on the borders are easily connected via a local bus wire. Bit-slice structures of arbitrary bit-width can be composed and then concatenated to form barrel shifters tailored to specific design needs.

The implementation in Figure 1 is faster because it contains fewer multiplexers. The critical path of the second implementation has more than twice as many MUX cells in series. The a wire cell, so the delay through three wire cells is approximately equal to the delay through one multiplexer cell. With both implementations being approximately the same size, and the critical paths containing about the same quantity of cells, the path with the lesser amount of multiplexers will be faster.

Figures 1 and 2 reflect the relative physical placement of the logical cells that compose the barrel shifter function and the busing structure that performs the interconnection.

Table 1 gives performance and utilization statistics for both implementations. Both implementations are available in schematic and layout form.

Figure 2. Modular Barrel Shifter with Non-coded Shift Control

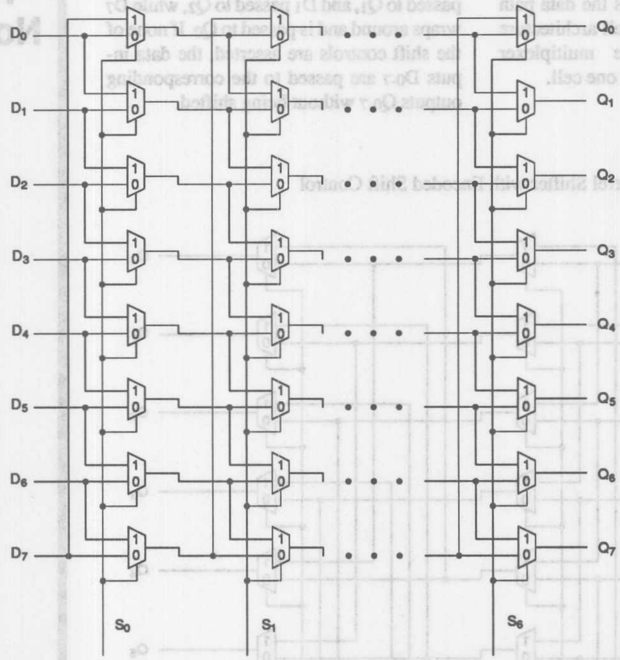


Table 1. Statistics for Barrel Shifter

Version	Cell Count(1)	Minimum Bounding Box (X x Y)	Maximum Speed(2)
Fast (3 Control Bits, Encoded)	112	8 x 14	28 ns / 35.7 MHz
Modular (7 Control Bits, Encoded)	126	13 x 10	44 ns / 22.7 MHz

Notes:
1. Includes cells used as wires.
2. $D_{0-7} \rightarrow Q_{0-7}$. Worst-Case Commercial Operating Conditions: 70°C, 4.75 V.

24-Bit Magnitude Comparator with 50 ns Response

Field Programmable Gate Array

Application Note

Introduction

The AT6000 Series field programmable gate array (FPGA) lets the designer implement a magnitude comparator that can compare two 24-bit binary integers in 50 ns.

Description

Figure 1 shows the structure of the magnitude comparator. Given two numbers $A_{0-23} > B_{0-23}$, the output GT will become asserted. If A_{0-23} is less than or equal to B_{0-23}, GT remains unasserted.

The logic necessary to compare two numbers can be derived iteratively according to four equations. The first two determine the largest number:

1. $T_n = A_nB_n'$

   For $n = 0$, where n is the significant bit.

2. $T_n = A_nB_n' + T_{n-1}(A_n + B_n')$

   For $n = 1$ *to* 23, where n is the significant bit.

The second two determine if the numbers are equal:

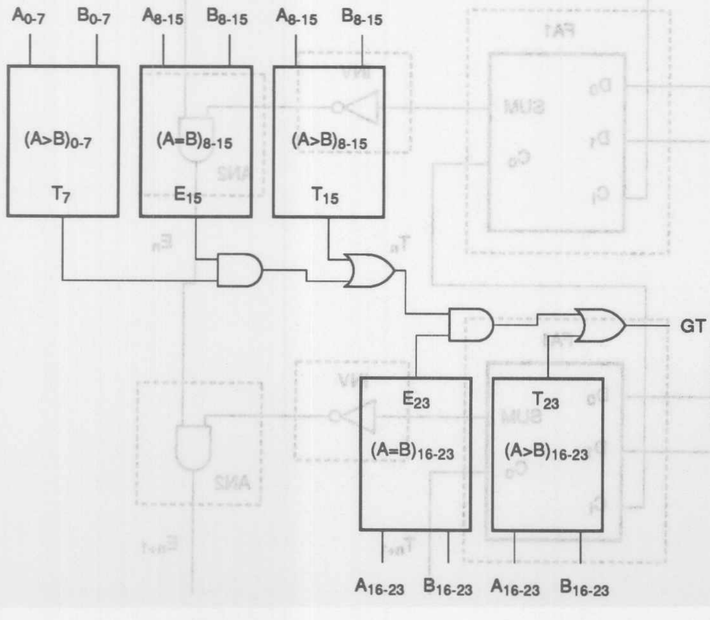
3. $E_n = A_n'B_n + A_nB_n'$

   For $n = 0$, where n is the significant bit.

4. $E_n = (A_n'B_n + A_nB_n') + E_{n-1}$

   For $n = 1$ *to* 23, where n is the significant bit.

Figure 1. Structure of 24-Bit Magnitude Comparator



6

Equation T_n could be used exclusively to generate the logic for GT, but the circuit would have a delay equivalent to 48 two-input NAND (ND2) gates for the worst-case comparison conditions.

To improve performance, the comparator is partitioned into three stages that each compare 8-bit portions of the two numbers. Thus, the parallel comparison of the three 8-bit portions of both numbers is faster than a single 24-bit implementation. The delay through an 8-bit stage is equivalent to 16 two-input NAND (ND2) gates.

Within each stage, the circuitry performs a bit-wise comparison starting between the most significant A_n and B_n bits within each stage, and asserts stage $T_{(m+1)(N/3)-1}$ (where stage m is 0 through 2, and $N = 24$ the bit-width of the compared numbers), if the most significant A_n is asserted and B_n is unasserted ($A_n > B_n$). If the most significant A_n and B_n bits are equal, then a pair of lesser significant bits within the stage must determine if the magnitude of the aggregate quantity of A bits is greater than the B bits in stage m. Also, the circuitry within each stage performs a bit-wise comparison on each pair of bits and asserts $E_{(m+1)(N/3)-1}$ if the 8-bit portions are equivalent.

Figure 2. Implementation of Magnitude Comparator

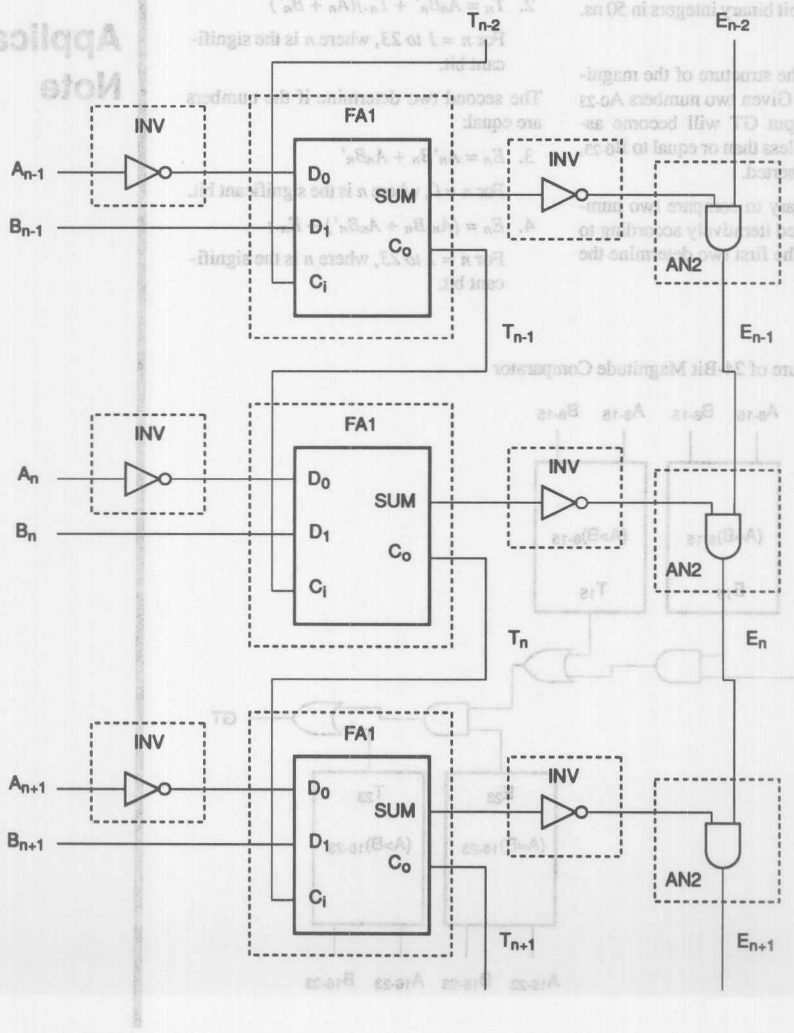
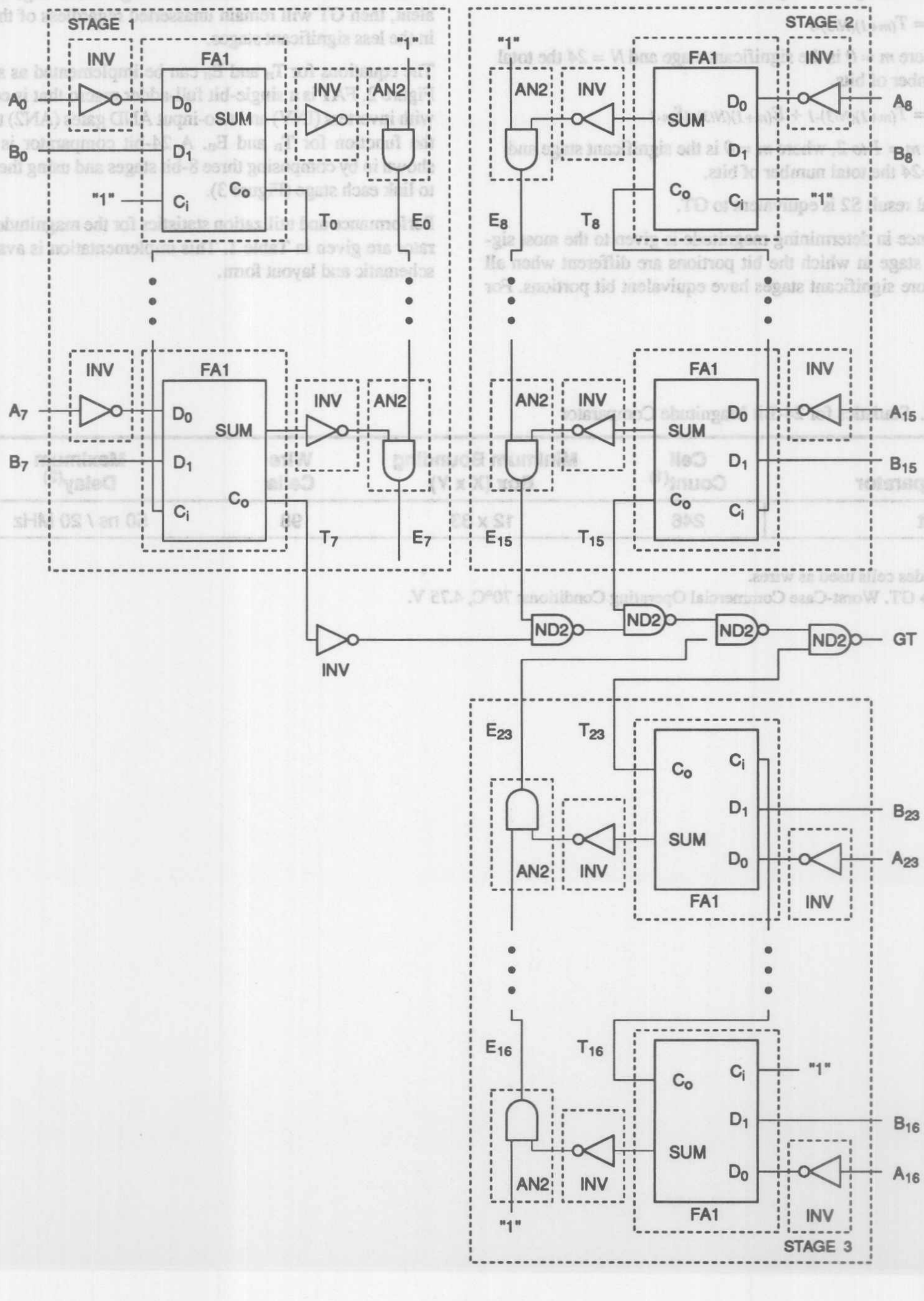


Figure 3. Magnitude Comparator is Composed of Three 8-Bit Stages



6

The $T_{(m+1)(N/3)-1}$ and $E_{(m+1)(N/3)-1}$ outputs from each 8-bit stage are fed into some logic to derive GT. This logic can be determined according to the equation:

1. $S_m = T_{(m+1)(N/3)-1}$

   Where $m = 0$ is the significant stage and $N = 24$ the total number of bits.

2. $S_m = T_{(m+1)(N/3)-1} + E_{(m+1)(N/3)-1}S_{m-1}$

   For $m = 1$ to 2, where $m = 0$ is the significant stage and $N = 24$ the total number of bits.

The final result S2 is equivalent to GT.

Precedence in determining magnitude is given to the most significant stage in which the bit portions are different when all other more significant stages have equivalent bit portions. For example, if T_{15} and E_{15} become unasserted, which means that the B bits are of greater magnitude than the A bits, and E_{23} asserts that the A and B bits in its more significant stage are equivalent, then GT will remain unasserted regardless of the results in the less significant stages.

The equations for T_n and E_n can be implemented as shown in Figure 2. FA1 is a single-bit full-adder macro that is combined with inverters (INV) and two-input AND gates (AN2) to realize the function for T_n and E_n. A 24-bit comparator is built as shown in by composing three 8-bit stages and using the S_n logic to link each stage (Figure 3).

Performance and utilization statistics for the magnitude comparator are given in Table 1. This implementation is available in schematic and layout form.

Table 1. Statistics for 24-Bit Magnitude Comparator

Comparator	Cell Count(1)	Minimum Bounding Box (X x Y)	Wire Cells	Maximum Delay(2)
24-Bit	246	12 x 33	98	50 ns / 20 MHz

Notes:
1. Includes cells used as wires.
2. $A_0 \rightarrow$ GT. Worst-Case Commercial Operating Conditions: 70°C, 4.75 V.

Field Programmable Gate Array

Application Note

16-Bit Four-to-One Multiplexer with 15-ns Delay

Introduction

The AT6000 Series field programmable gate array (FPGA) lets the designer implement a 16-bit, four-to-one multiplexer with a 15-ns delay from the select control to the most significant output bit. Performance is enhanced by a unique feature of the busing architecture that enables the select control lines to be distributed across the multiplexer data path with minimal skew.

Description

Figures 1 and 2 show the relative physical placement of the logical cells that compose the multiplexer function and the busing structure that performs the interconnection. Four 16-bit input buses, A_{0-15}, B_{0-15}, C_{0-15}, and D_{0-15}, can be multiplexed to a 16-bit output bus F_{0-15}. Two select lines S_0 and S_1 determine which of the input buses are multiplexed to the output. Figure 1 shows several four-to-one multiplexer bits, each implemented in four cells. A cell can be configured as a two-to-one multiplexer (MUX21). Every MUX21 is tied to either the S_0 or S_1 select lines. S_0 controls 32 MUX21 cells, and S_1 controls 16 MUX21 cells.

Typically, distribution of a signal to such a large number of cells would cause unacceptable skew between the cells closest and furthest from the source. By aligning the multiplexers, and routing a single wire to every one, the most efficient signal distribution method is realized. However, the load on the wire might unacceptably decrease its slew rate. Buffering the wire after routing it to a certain number of multiplexers would improve the slew rate, but additional cells would be required for the buffering. Skew would also be introduced as buffer stages are added to the distribution network. Other signal distribution methods have been shown to have minimal skew and optimal slew rate, but they require more logic resources.

Using express buses, skew is minimized and slew is improved by segmenting the load along each signal. Figure 2 shows that the express and local buses can distribute S_0 and S_1 without using additional logic cells. S_0 passes from a local bus into a repeater—a programmable cross-bar buffer—and is routed onto the express bus. At each repeater stage, taps are taken off onto the local bus and into the multiplexers. Since the express bus is essentially an unloaded wire, its delay is much less than that of the local bus, which is loaded by multiplexer select inputs. Although signal propagation of S_0 and S_1 is serialized from the source output through several repeater stages to the final inputs, this distribution method is the most efficient for performance and cell utilization.

Table 1 gives performance and utilization statistics for the multiplexer. It is available in schematic and layout form.

6

Figure 1. Schematic of Multiplex Function for Several Output Bits

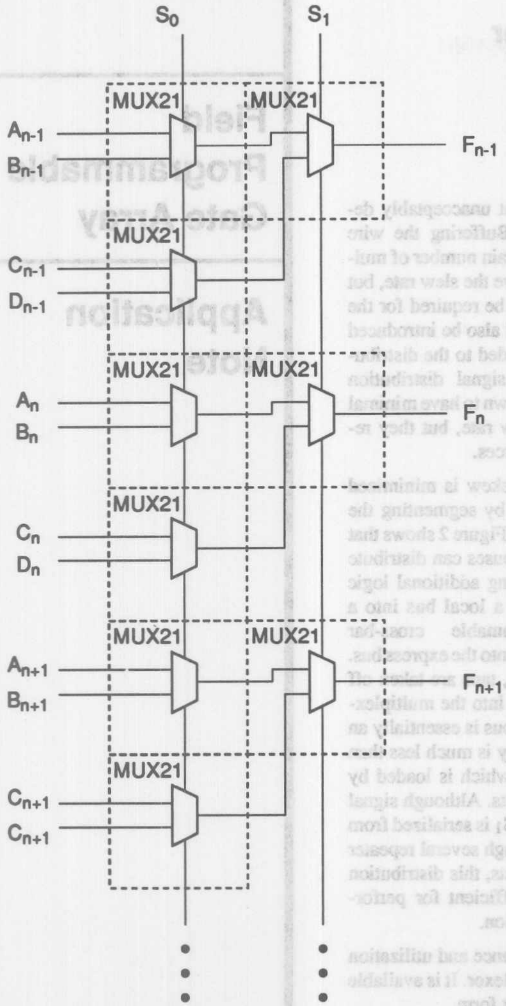


Figure 2. Equivalent Layout with Busing Structure

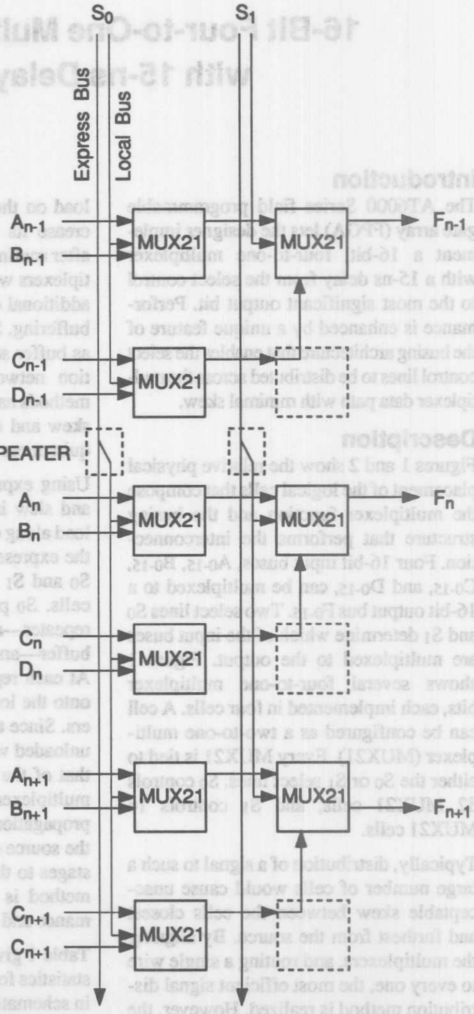


Table 1. Statistics for Four-to-One Multiplexer

Multiplexer	Cell Count(1)	Minimum Bounding Box (X x Y)	Maximum Delay(2)
16-Bit	64	2 x 32	15 ns / 66.7 MHz

Notes:
1. Includes cells used as wires.
2. $S_0 \rightarrow F_{15}$. Worst-Case Commercial Operating Conditions: 70°C, 4.75 V.

Field Programmable Gate Array

Application Note

8-Bit, S-P/P-S "Corner-Bender" Data Converter

Introduction

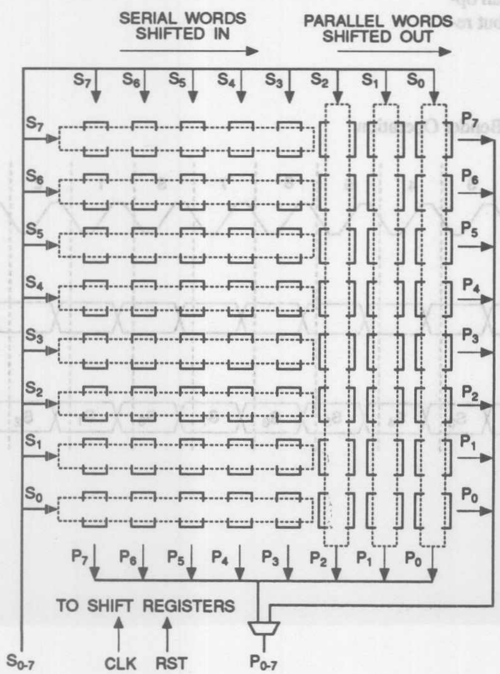
With the proliferation of computer and voice networks that carry digitized analog data, data conversion applications have become commonplace. For example, the use of time-division multiplexing in broadcasting and receiving circuitry requires fast serial-to-parallel (S-P) and parallel-to-serial (P-S) data conversion. Using the AT6005 device, two S-P/P-S corner-bender circuits were implemented: one optimized for area and power consumption, the other for speed and expandability.

Description

Figure 1 is the functional representation for an 8-bit corner-bender. CLK is the clock input signal and RST is the asynchronous reset. RST is active low. Pins S_0 through S_7 are the data inputs; pins P_0 through P_7 are the converted data outputs.

The corner bender accepts 8-by-8 blocks as input. Each block is eight bits wide and arrives serially over eight clock cycles. The corner bender transforms each input block into an 8-by-8 output block such that bits aligned in parallel in the input block are aligned serially in the output block, and vice versa.

Figure 1. Functional Representation of Corner Bender



6

The two implementations—low-area/low-power and high-speed/expandable—are functionally identical. Initial power-up of the AT6005 device resets all user registers, so the circuit begins data conversion on the first rising edge of CLK after configuration.

In serial-to-parallel mode, the $P_{0\text{-}7}$ outputs remain low for the first seven clock cycles. After the rising edge of the eighth clock cycle, the first 8-bit serial data word from S_0 is available at the $P_{0\text{-}7}$ outputs. The S_1 serial word is available after the rising edge of the next clock (Figure 2). Subsequent serial words are available on the rising edge of every clock until RST is asserted. Asserting RST at any time clears the registers and inhibits the conversion.

The first implementation minimizes area and power consumption, but still operates at up to 22 MHz in the AT6005-4 and 31 MHz in the AT6005-2.

Figure 3 shows the gate-level equivalent of the register block structure used for conversion. A single AT6005 cell is configured as a two-input multiplexer feeding a D-type flip-flop (FDMUX).

After the rising edge of every eighth clock cycle, two-to-one multiplexers switch the direction of the data flow into the register from west-to-east to north-to-south. The boundary registers that feed the outputs, $P_{0\text{-}7}$, are also switched after the rising edge of every eighth clock cycle. Pins $P_{0\text{-}7}$ receive parallel (or serial) data from the boundary registers on the axis perpendicular to, and on the opposite side of, the serial (or parallel) data input.

The second implementation maximizes throughput and can operate at 52 MHz (AT6005-4) and 75 MHz (AT6005-2), but requires almost twice the area and 25 percent more power per MHz than the first.

A secondary set of registers is used to latch a parallel (or serial) 8-bit word from a serial (or parallel) data register on the rising edge of every eighth clock cycle. Figure 4 shows the gate-level equivalent of the register block structure used for the conversion. The shift registers employ D-type flip-flops (FD); the secondary registers use FDMUX macros.

At the rising edge of every eighth clock cycle, the secondary registers load a parallel (or serial) 8-bit data word from the serial (or parallel) shift registers. On subsequent clock cycles, the parallel (or serial) words are shifted from each set of secondary registers until the words reach $P_{0\text{-}7}$.

The slower implementation requires fewer cells and uses less area, but has a longer critical path delay and hence less throughput than the faster implementation. In the first implementation, the longest interconnection occurs between the registers containing the most significant bit, S_0, which feeds the two-to-one multiplexer driving P_7, MUX21 (Figure 5).

The longest path between the register and multiplexer spans nearly half the circumference of the layout, which is four times longer than the longest path in the second implementation. Each flip-flop in the second circuit connects to only its two nearest neighbors, making the interconnects fairly equal in length. As a result, the second implementation can be expanded without significant performance loss. Table 1 gives a comparison of the performance and utilization statistics for the two corner benders. Both circuits are available in schematic and layout form.

Figure 2. Timing Diagram of Serial-to-Parallel Corner-Bender Operation

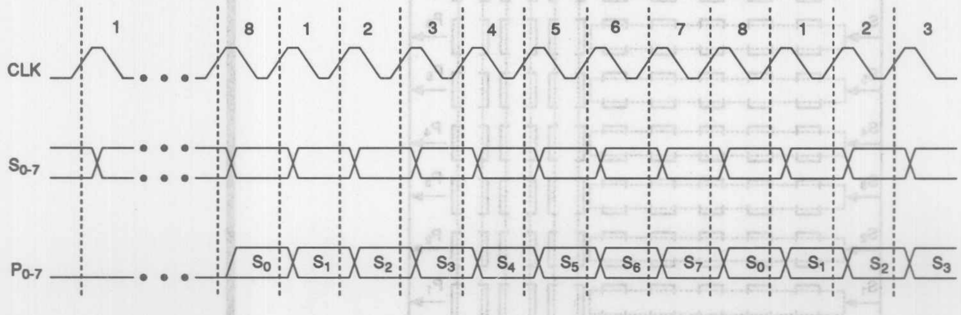


Figure 3. Schematic of Low-Area/Low-Power Core

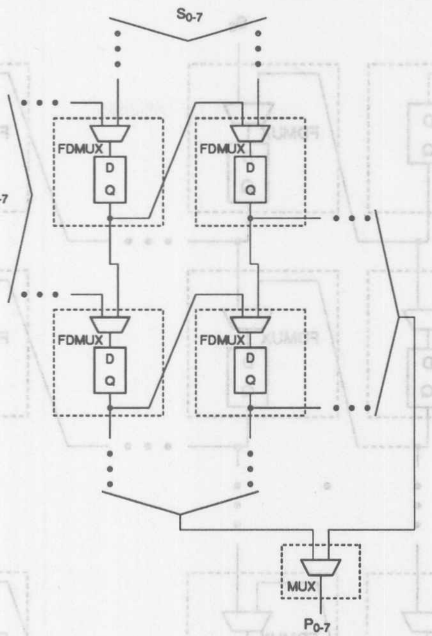
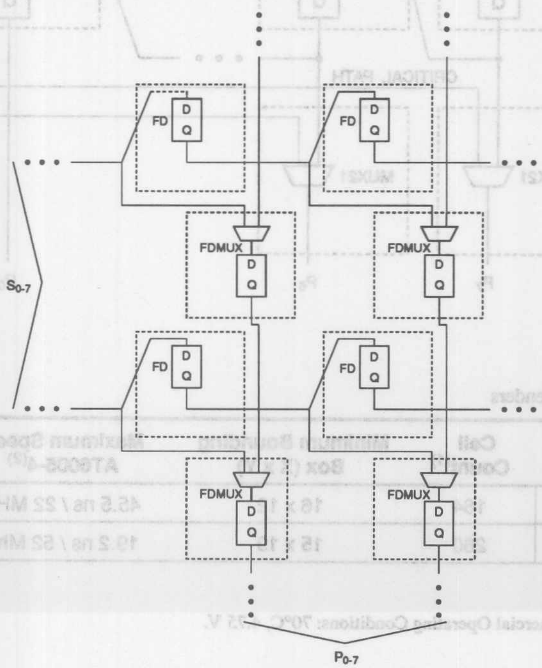


Figure 4. Schematic of High-Speed/Expandable Core



6

Figure 5. Interconnect Scheme of First Implementation

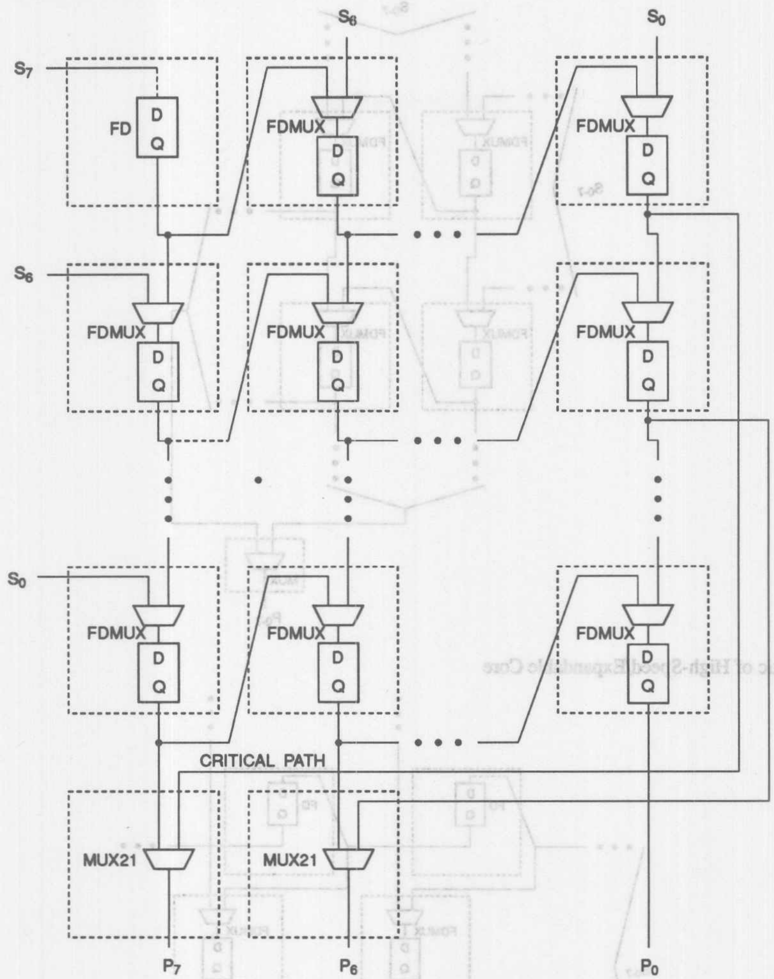


Table 1. Statistics for Corner Benders

Corner Bender	Cell Count[1]	Minimum Bounding Box (X x Y)	Maximum Speed AT6005-4[2]	Maximum Speed AT6005-2[2]
Low Area/Low Power	164	16 x 12	45.5 ns / 22 MHz	32.3 ns / 31 MHz
High Speed/Expandable	260	15 x 19	19.2 ns / 52 Mhz	13.3 ns / 75 MHz

Notes:
1. Includes cells used as wires.
2. CLK $\rightarrow P_o$. Worst-Case Commercial Operating Conditions: 70°C, 4.75 V.

Field Programmable Gate Array

Application Note

16-Word by 8-Bit FIFO

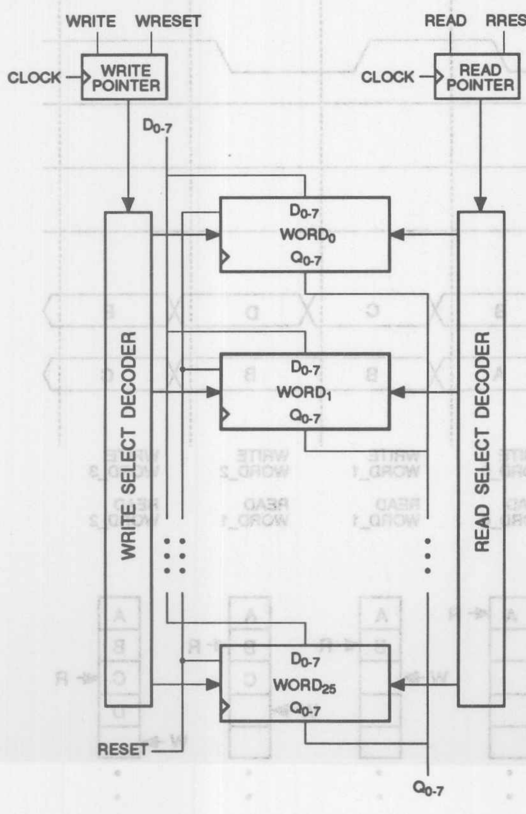
Introduction

The AT6000 Series field programmable gate array (FPGA) lets the designer implement a synchronous first-in, first-out (FIFO) register buffer with a word width and depth tailored to specific design needs. A 16-word FIFO with each word being eight bits is constructed and analyzed in the AT6005-2 device, and shown to have 15 MHz performance

Description

Figure 1 shows a functional block diagram of the FIFO architecture. Data on inputs D_{0-7} is latched into a particular word register on the rising edge of CLOCK when WRITE is asserted. When READ is asserted, the outputs Q_{0-7} are driven by a new word register. The write and read pointers, together with the select decoder logic, determine the particular register or registers that are the target of the write operation and the source of

Figure 1. Architecture of 16 x 8 FIFO

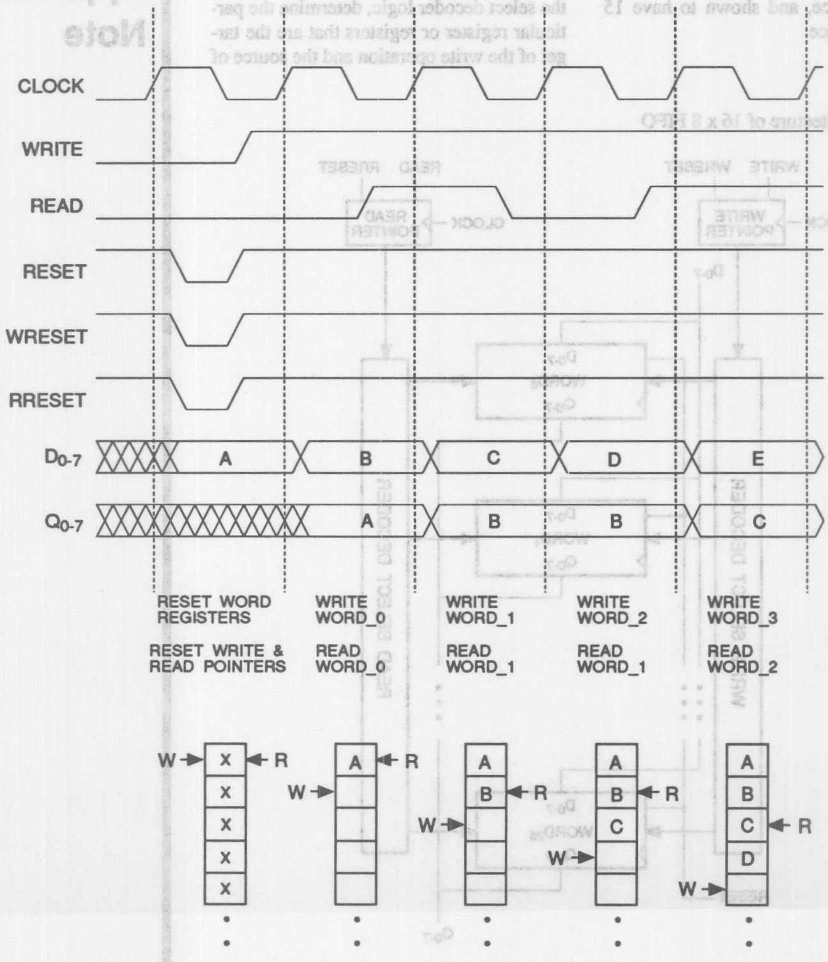


6

the read operation. Individual read or write operations can be executed every clock cycle when either the READ or WRITE signals are asserted. Simultaneous write and read operations are also possible to any one word register or combination of two registers. If neither the READ or WRITE signals are asserted, the pointers maintain their current address. After a write operation occurs, the write pointer is automatically updated to point to the next word register, and then awaits the next write operation. The read pointer allows the data of a particular word register to be continuously available until the next read operation, after which new data from the next word register is accessed. Essentially, the read pointer points to the last word that was read and the write pointer points to the next word to be written.

The RESET, WRESET, and RRESET signals control the initialization of various portions of the FIFO. When RESET is set low, the word registers are immediately cleared. RESET is an asynchronous signal that causes all bits within each word register to be set low. WRESET and RRESET are synchronous initialization signals for the write and read pointers. If WRESET is taken low and then high before the rising edge of CLOCK, the write pointer is set to the first word register. A write operation can then commence on the next rising edge of CLOCK if WRITE is asserted. The RRESET signal controls the read pointer in the exact same manner. Figure 2 shows the operation of the FIFO from the relative timing of the control signals.

Figure 2. Relative Timing of FIFO Control Signals



A schematic of the word registers is shown in Figure 3. Each bit of the data word is stored in an FDMUX macro, which is a two-to-one multiplexer feeding a D-type flip-flop. If the word register is selected for a write operation, the two-to-one multiplexer chooses $D_{0\text{-}7}$ as the input to be latched into the D-type flip-flop on the next rising edge of CLOCK. If the word register is not chosen for a write operation, the two-to-one multiplexer recirculates the data value already present in the D-type flip-flop. When a READ operation is initiated, the logic values present at the D-type flip-flop outputs are passed through the tri-state output buffers BUFZ onto an internal tri-state bus that ties together the output of every word register. Since the inputs $D_{0\text{-}7}$ and outputs $Q_{0\text{-}7}$ enter and exit on different wires, simultaneous write and read operation can occur on the same clock cycle.

Incorporated into each word register is a portion of the read and write pointer logic. The write pointer is a controlled shift register that is 16 bits in length. Upon initialization the first bit in the shift register is set to a logical "one" value, and all other bits are reset low. After a write operation occurs to the first word register, the "one" is passed to the next bit in the write pointer shift register at the rising edge of CLOCK. This "one" bit will continue to loop around the shift register whenever WRITE is asserted upon the rising edge of CLOCK. As the "one" is passed along, it allows a write operation to occur to the word register. The read pointer is implemented in essentially the same manner.

In the AT6000 architecture, implementing the write and read pointers as controlled shift registers has several advantages over controlled counter/decoder methods. For example, several 5-bit modular 16 counters together with several 5-bit to 16-bit decoders could be used to implement the read and write pointers in a controlled counter/decoder approach. Although the counters would use far less flip-flops, the decoder logic would be five times as large as in the controlled shift register method. More control signals would also have to be bused to every word register since in the controlled counter/decoder method every output bit of the counters must be used as a select control for the decoders of both the read and write pointers. The additional decoder logic and busing necessary for the controlled counter/decoder makes it ostentatiously large. Since every cell in the AT6000 architecture contains a D-type flip-flop, the controlled shift register approach is more efficient than the controlled counter/decoder approach.

Figure 3. Modular Word Register Schematic

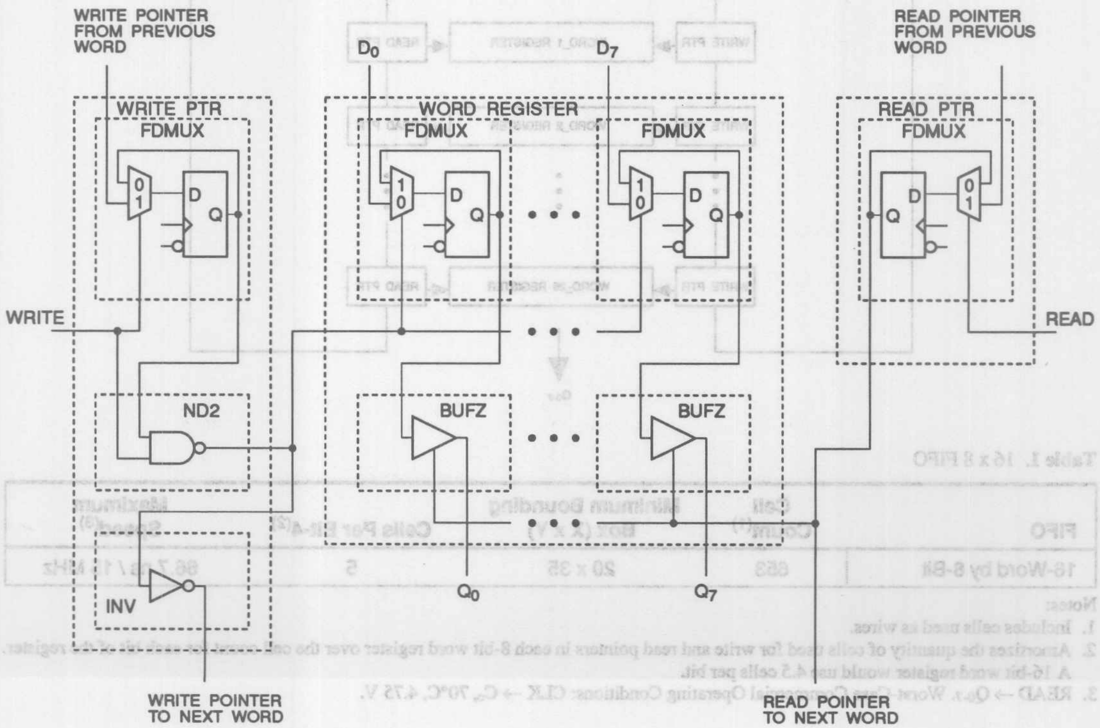


Figure 4 shows the initialization logic along with several word registers. The initialization logic synchronously allows either the write or read pointers to be individually or simultaneously reset to point to the first word register. The word registers can be asynchronously cleared. For this example, 16-word registers are linked together to form a 16-word FIFO, with each word being eight bits in length. The modular construction allows the concatenation of word registers to form a FIFO of any length.

The word registers can also be widened as long as the portions of the write and read pointers are maintained on either side. Only one unique register is needed to hold the initialization logic to mark the first word of the FIFO.

The performance and utilization statistics for the 16-word by 8-bit FIFO are given in Table 1. This implementation is available in schematic and layout form.

Figure 4. Architecture of 16-Word x 8-Bit FIFO

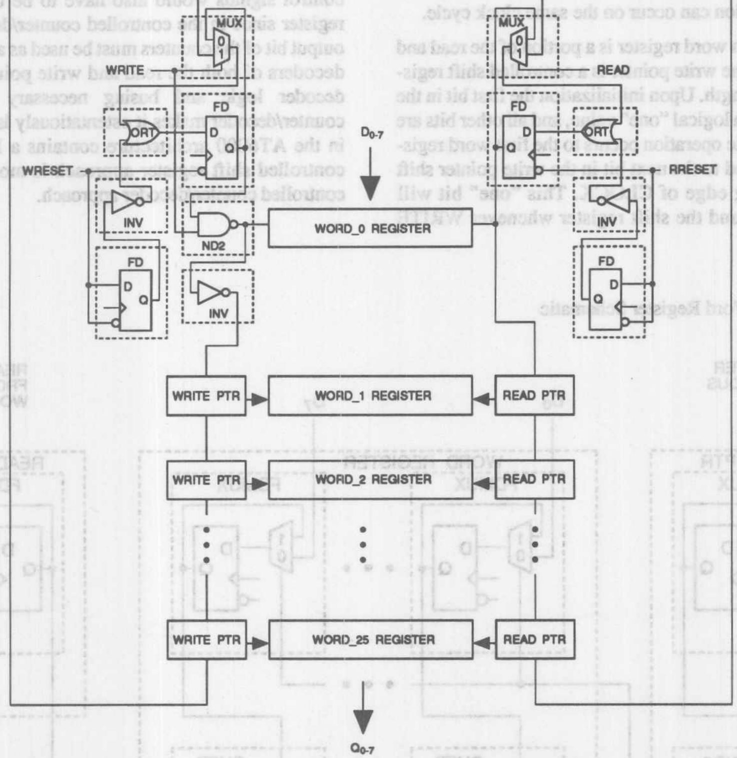


Table 1. 16 x 8 FIFO

FIFO	Cell Count(1)	Minimum Bounding Box (X x Y)	Cells Per Bit-4(2)	Maximum Speed(3)
16-Word by 8-Bit	653	20 x 35	5	66.7 ns / 15 MHz

Notes:
1. Includes cells used as wires.
2. Amortizes the quantity of cells used for write and read pointers in each 8-bit word register over the cell count for each bit of the register. A 16-bit word register would use 4.5 cells per bit.
3. READ → Q_{0-7}. Worst-Case Commercial Operating Conditions: CLK → C_o, 70°C, 4.75 V.

IEEE 1149.1-1990 Standard Test Access Port and Boundary-Scan

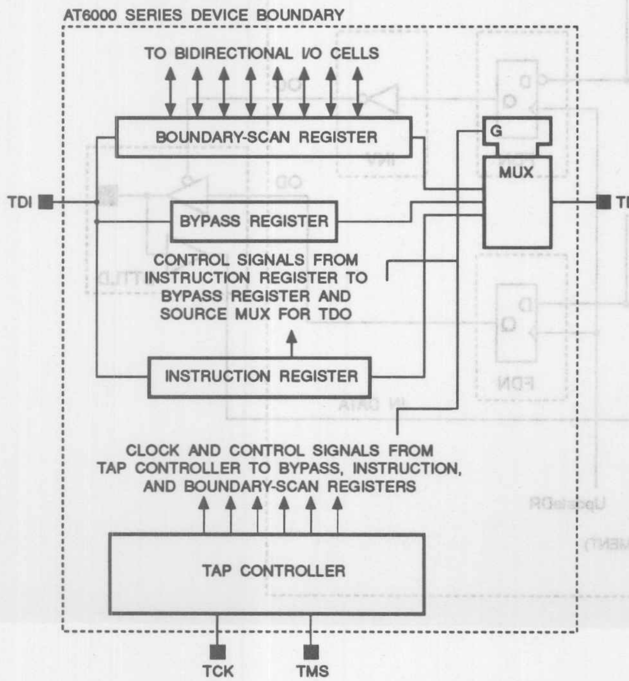
Field Programmable Gate Array

Application Note

Introduction

For system or board diagnostics, AT6000 Series devices can be programmed with the 1149.1 standard test logic and then reprogrammed for normal operation when the diagnostics are complete. The area and performance overhead of the test logic does not impact normal operation in the device because it is replaced by the logic for normal function. All mandatory test instructions can be executed with this portable test logic configuration, which guarantees conformance to the 1149.1 standard. The 1149.1 standard provides a consistent mechanism for confirming that each component in a system performs its required function. Since AT6000 devices are factory- tested with test patterns that exercise all the programmable features, integrity is insured without having to rely on standard test logic. Standard test logic is best suited for checking the interconnections between devices on the board. Boards are often multi-layer and double-sided, making traditional board test methods, like the bed-of-nails approach, expensive and impractical.

Figure 1. Block Diagram of 1149.1 Test Logic Architecture



Recommended Test Procedure

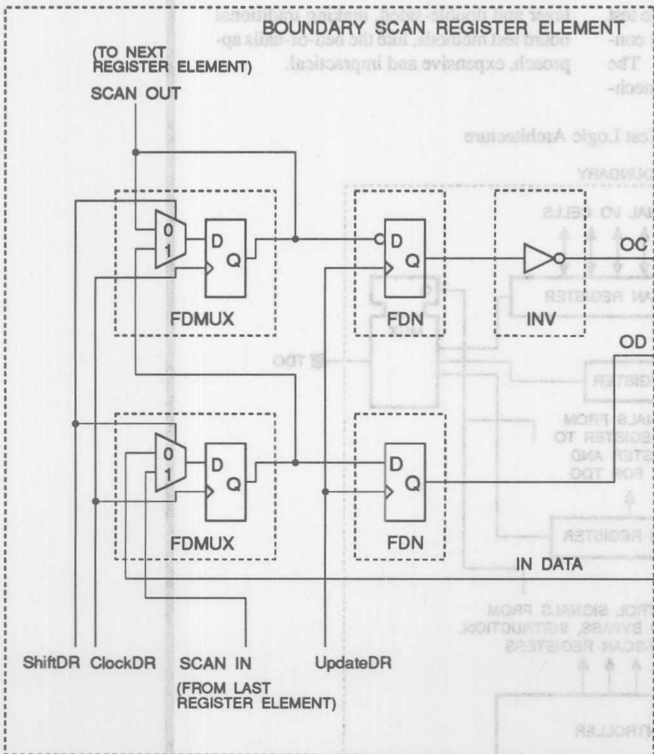
Figure 1 shows the test logic architecture with dedicated I/O pins TCK, TMS, TDI, and TDO for the 1149.1 standard. TCK and TMS control the Test Access Port controller, a state machine that regulates the flow of serial test patterns and results from TDI to TDO. The boundary-scan register communicates with all usable I/O pins, which are configured as bi-directional drivers. The bypass register enables data to be moved more quickly through the device by bypassing the boundary-scan register. The instruction register holds the test instruction that activates either the boundary-scan or bypass register, and also helps control the MUX that supplies the source for TDO.

The recommended procedure for using 1149.1 Boundary Scan is to program the part with the test logic during a system diagnostics routine, then initiate boundary scan testing for the system or board. When the test is done, reload the AT6000 device with system logic for normal operation. This test method eliminates the optional test capabilities defined in the standard. For example, using the SAMPLE/PRELOAD test instruction to capture the state of the I/O for on-line diagnostics during normal operation is not possible since the test logic is replaced with system logic.

Other optional instructions facilitating internal fault testing and the execution of built-in self testing procedures (BIST) also cannot be supported with this test methodology.

For example, INTEST isolates the device, allows patterns to be serially shifted in and applied to the internal logic, then captures the output results within the device so that they can be shifted out for fault analysis. RUNBIST enables proprietary, built-in self-test procedures to be initiated by the standard test logic. The intended test capabilities of both these instructions are already covered by the factory testing performed on each AT6000 device. Therefore, eliminating these instructions and their associated overhead by loading and unloading the standard test logic does not increase the risk of overlooking a fault in the device.

Figure 2. Boundary Scan Cell



Description of Boundary Scan Operation

As shown in Figure 2, the boundary-scan register element shifts data in through Scan In on the rising edge of ClockDR when ShiftDR is asserted. Since every I/O pin is configured as a bi-directional driver during the test sequence, two shift register bits are needed to control the output signal OD and the output control OC of the bi-directional driver. Each bit in the shift register is then latched to a secondary register that directly drives an output pin or output control of the bi-directional driver. UpdateDR loads the secondary registers from the shift register. If the bi-directional I/O BTTLD is set as an input (OC=1), the signal present at the pin is latched into the shift register on the rising edge of ClockDR when ShiftDR is unasserted.

A mandatory 1149.1 requirement for programmable components is that the length of the boundary-scan register be independent of the way the component is programmed in normal system operation. Configuring every I/O to be bi-directional eliminates the dependency of the test logic on the particular I/O pin assignment utilized during normal operation. The test logic has the flexibility to set the direction of any I/O pin according to the patterns shifted in and transferred to the secondary registers that drive the output control pins. These patterns are formulated by the test engineer based on specific knowledge about the I/O pin assignment of the device during normal operation, and how the device is interconnected with other components on the board.

These secondary registers hold a pattern at the output pins while the results are sampled at the chip inputs and stored in the shift register. ShiftDR controls sampling and shifting of the data in the shift register. The secondary registers hold the output control pins stable during sampling and shifting out of the result for interpretation because the values in the shift register change. If the secondary registers were eliminated, then the shift-register element might incorrectly drive the output control during data shifting. More than one output from different devices might accidentally drive the same wire for a substantial period of time, which could damage the devices. Having a secondary register to hold the bi-directional output control signals during shifting prevents this.

Sampling and shifting is controlled by a synchronous state machine within the device. Control signals ClockDR, UpdateDR, and ShiftDR originate from this state machine, called a test-access-port (TAP) controller. External TAP signals—which might originate from automated test equipment (ATE) or a diagnostic processor—enter the capitalize controller through dedicated input pins and initiate state-to-state transitions. The following TAP signals go into the controller:

TCK: Test Clock input provides an independent clock signal for the test logic (i.e., boundary-scan register, instruction register, bypass register, and state machine)

TMS: Test Mode Select input controls the transitions of the TAP state machine, which determines when data is sampled, loaded, and shifted.

TDI: Test Data Input is the serial data input into the shift register for the test pattern.

TDO: Test Data Output is the serial data output from the shift register.

The signals TDI and TDO, although considered part of the TAP, do not initiate or affect the state-to-state transitions in the TAP controller. They provide the serial link to and from other components on the board. For a detailed state diagram of the TAP controller, refer to Chapter 5 of the JTAG (Joint Test Action Group) standard.

Composing a Test Pattern for the AT6005

The test pattern has two purposes: first, to set the I/O as either inputs to or outputs from the device, and second, to provide a logical value ("1" or "0") to the output drivers from the device. The designer or test engineer is responsible for composing the pattern so that the proper inputs and outputs will be present on the device boundary. By performing the following tasks, any designer or test engineer somewhat familiar with the 1149.1 standard can prepare the patterns for boundary-scan testing:

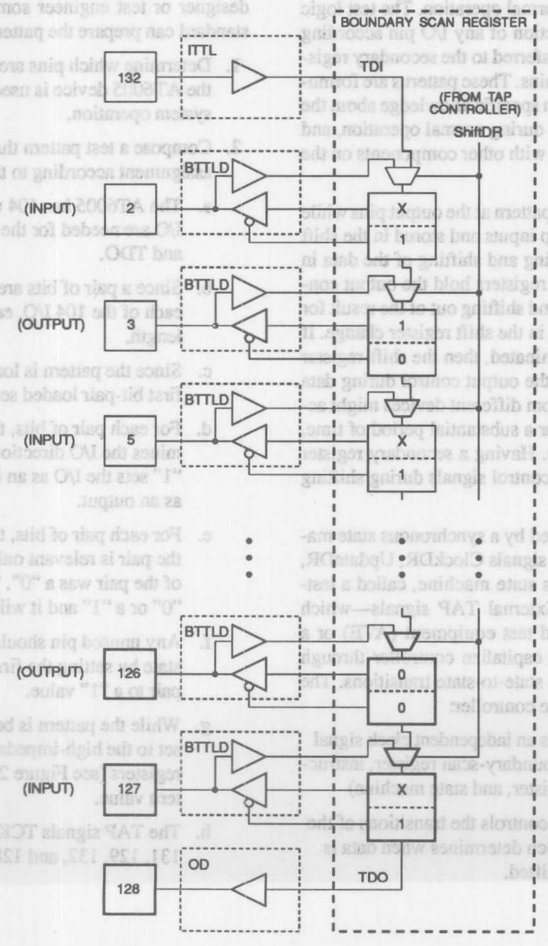
1. Determine which pins are inputs and outputs based on how the AT6005 device is used in the board during normal system operation.
2. Compose a test pattern that will set the I/O to match the pin assignment according to the following assumptions:
   a. The AT6005 has 104 usable I/O because four dedicated I/O are needed for the TAP signals TCK, TMS, TDI, and TDO.
   b. Since a pair of bits are needed to set the direction of each of the 104 I/O, each pattern must be 208 bits in length.
   c. Since the pattern is loaded serially starting at TDI, the first bit-pair loaded sets the last I/O pins.
   d. For each pair of bits, the first serially loaded bit determines the I/O direction (input or output). A logical "1" sets the I/O as an input, a logical "0" sets the I/O as an output.
   e. For each pair of bits, the second serially loaded bit of the pair is relevant only if the first serially loaded bit of the pair was a "0". The second bit can be either a "0" or a "1" and it will drive the output pin.
   f. Any unused pin should be driven to a high-impedance state by setting the first bit of its corresponding bit-pair to a "1" value.
   g. While the pattern is being loaded, the output pins are set to the high-impedance state until the secondary registers (see Figure 2) are updated with the test pattern value.
   h. The TAP signals TCK, TMS, TDI, and TDO use pins 131, 129, 132, and 128, respectively.

Figure 3 shows a test pattern in the boundary-scan register and its effect on the BTTLD bi-directional I/O. Pin 2 is set as an input because the output enable signal of BTTLD is set to "1." Since the I/O is set as an input, the value of the first bit in the corresponding bit-pair in the boundary-scan register that drives the output buffer portion of BTTLD really does not matter (it is represented as an "X"). Pins 5 and 127 are also set as inputs because their output enable bits for BTTLD are also set as "1" in the boundary-scan register.

If the second bit of the bit-pair that controls the BTTLD I/O macros were a "0," as is the case with Pins 3 and 126, the BTTLDs are set as outputs. The outputs of BTTLD for Pins 3 and 126 are both driven by the first bit of the corresponding bit-pair in the boundary-scan register. A "1" from the boundary-scan register drives Pin 3, and a "0" drives Pin 126.

A test pattern is serially shifted in through Pin 132. For example, the first bit-pair is shifted through the entire boundary-scan register until it is in position to set the BTTLD macro of pin 127. While the pattern is being shifted through the register, the outputs of the secondary registers that the drive the output enable pins of the BTTLD macros are set to "1" so that every I/O in the device is in the high-impedance state (see Figure 2). This precautionary measure prevents device pins from driving a signal onto a board wire that might contend with a signal from another chip. When the pattern is completely shifted into the boundary-scan register, the secondary registers are updated with the pattern, thus setting the direction of the BTTLD macros.

Figure 3. Test Pattern in Boundary-Scan Register

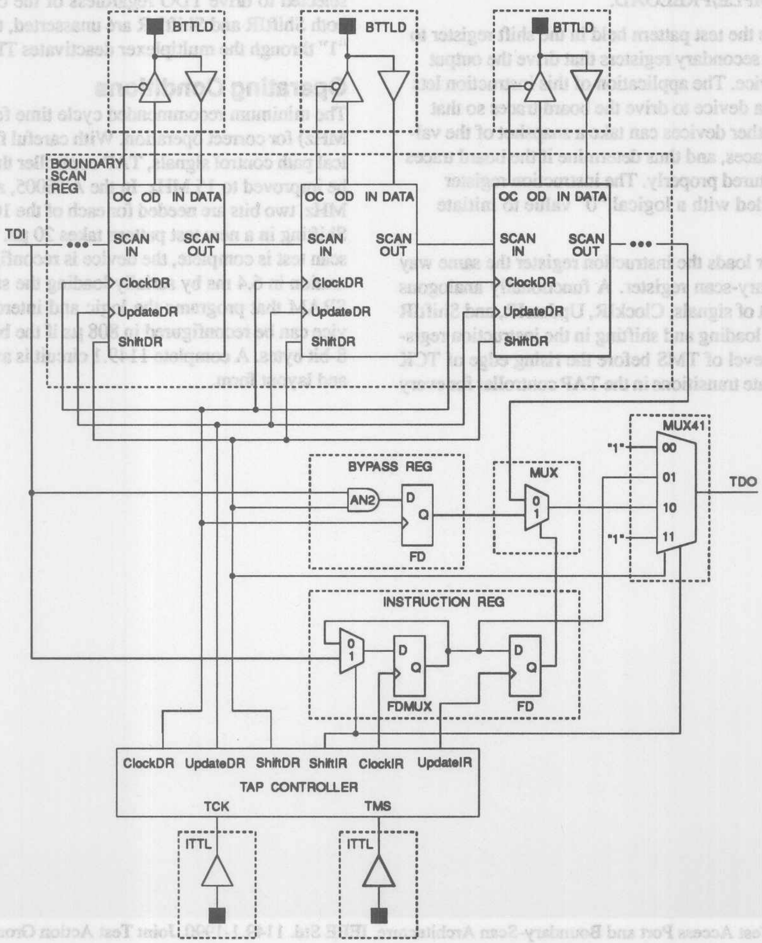


Test Logic Architecture

Figure 4 shows a more detailed schematic of the test logic. The TAP controller performs boundary-scan by first capturing signals at the input pins of the bi-directional drivers and loading them into a corresponding shift register element. The TAP controller sets ShiftDR low and takes ClockDR from low to high to perform the capture. The captured input signals originate from the outputs of other chips on the board as the result of a previous test pattern. Shifting the captured pattern out of the device occurs when the TAP controller sets ShiftDR high and toggles ClockDR. While the result is being shifted out through TDO, a new test pattern is simultaneously being shifted in through TDI. The new pattern is loaded into the secondary registers when the TAP controller toggles UpdateDR. Any bi-directional driver that is set as an output will then drive the board wires leading to the inputs of other chips.

Also shown in Figure 4 are bypass and instruction registers. The bypass register quickly passes test patterns through the device if the patterns are designated for another component on the board. Captured test pattern results also flow through the bypass register to hasten their arrival to another part of the diagnostics system. Depending on the instruction loaded into the instruction register, either the bypass register or the boundary-scan register provides the source for TDO. The logical value loaded into the instruction register controls the multiplexer MUX that selects either the boundary-scan or bypass register to drive TDO.

Figure 4. Schematic of 1149.1 Test Logic Architecture



6

The following mandatory instructions are necessary to conform to the standard (as described in Chapter 7 of the JTAG standard):

BYPASS: Causes a serial test pattern being shifted into TDI to be passed through a single bit register (called the bypass register) and then to TDO. The purpose of this instruction is to allow rapid serial movement of test patterns and results between components on a board. The instruction register should be load with a logical "1" value to initiate BYPASS.

SAMPLE/PRELOAD: Allows a snapshot of the data present at the input pins of a device to be stored in the shift register, and then allows the snapshot to be shifted out through TDO. While the data is being shifted out, a new test pattern can be simultaneously shifted in through TDI. The instruction register should be load with a logical "0" value to initiate SAMPLE/PRELOAD.

EXTEST: Allows the test pattern held in the shift register to be loaded into secondary registers that drive the output pins of the device. The application of this instruction lets the outputs of a device to drive the board traces so that the inputs of other devices can take a snapshot of the values on those traces, and thus determine if the board traces were manufactured properly. The instruction register should be loaded with a logical "0" value to initiate EXTEST.

The TAP controller loads the instruction register the same way it loads the boundary-scan register. A functionally analogous but independent set of signals, ClockIR, UpdateIR, and ShiftIR are used to control loading and shifting in the instruction register. Changing the level of TMS before the rising edge of TCK initiates state-to-state transitions in the TAP controller for every TCK cycle, causing the outputs of the controller (ClockDR, ShiftDR...) to change. A particular sequence of level changes in TMS over several TCK cycles results in the execution of either SAMPLE/PRELOAD or EXTEST if the value in the instructions register is a logical "0."

The TAP controller also chooses the register (boundary-scan, bypass, or instruction) that drives TDO. When the ShiftDR signal is asserted either the boundary-scan register or the bypass register is selected. The instruction register must then determine the register to drive TDO. If the instruction register is loaded with a "1," meaning that the BYPASS instruction should be executed, then the bypass register is selected. A logical "0" in the instruction register means that either the EXTEST or SAMPLE/PRELOAD should be executed, and that the boundary-scan register is selected.

When the ShiftIR signal is asserted the instruction register is selected to drive TDO regardless of the current instruction. If both ShiftIR and ShiftDR are unasserted, then driving a logical "1" through the multiplexer deactivates TDO.

Operating Conditions

The minimum recommended cycle time for TCK is 100 ns (10 MHz) for correct operation. With careful floor planning of critical path control signals, TAP controller the performance could be improved to 15 MHz. In the AT6005, at a TCK speed of 10 MHz, two bits are needed for each of the 104 bi-directional I/O. Shifting in a new test pattern takes 20 µs. When the boundary-scan test is complete, the device is reconfigured for normal operation in 6.4 ms by serially loading the stream of bits into the SRAM that programs the logic and interconnections. The device can be reconfigured in 808 µs if the bit stream is loaded as 8-bit bytes. A complete 1149.1 circuit is available in schematic and layout form.

References:

IEEE Standard Test Access Port and Boundary-Scan Architecture, IEEE Std. 1149.1-1990, Joint Test Action Group (JTAG), Institute of Electrical and Electronic Engineers (IEEE), New York, May 21, 1990.

Digital Frequency/Phase Comparator (DFPC)

Field Programmable Gate Array

Application Note

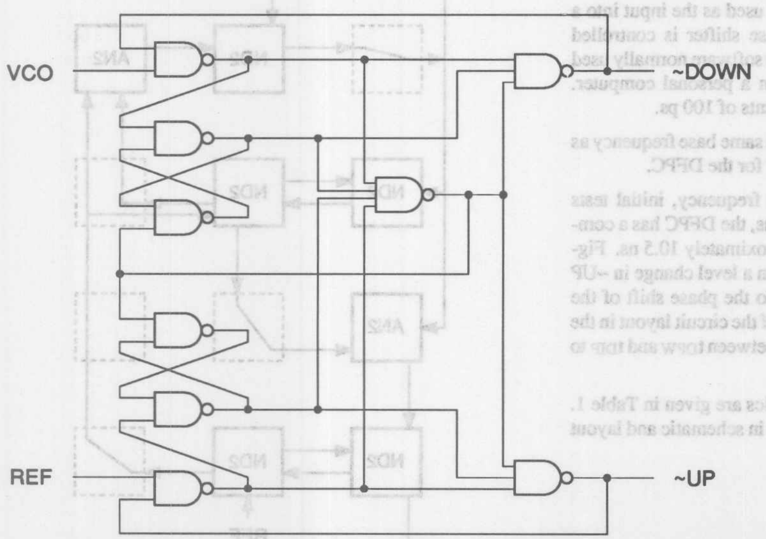
Introduction

The AT6000 Series field programmable gate array (FPGA) lets the designer implement a digital frequency/phase comparator (DFPC) that interfaces to a voltage controlled oscillator (VCO).

Description

The DFPC has two periodic input signals, a reference frequency of some desired value and a variable frequency from the VCO. Two output signals from the DFPC, ~UP and ~DOWN, control the VCO. These two signals change the VCO output frequency so as to match the reference frequency entering the DFPC. Once the reference and the variable signals have the same phase and frequency, the DFPC continues to make small adjustments to the VCO frequency to maintain the match.

Figure 1. Digital Frequency/Phase Comparator Circuit



A representation of the circuit is shown in Figure 1. The exact implementation of the circuit in the AT6000 architecture is shown in Figure 2. The performance of the DFPC was determined with the aid of another specially designed test circuit, which measures three performance characteristics of the DFPC:

1. The minimum phase difference in nanoseconds which the DFPC can detect between the reference and variable VCO signals;
2. The width of the common mode time when both ~UP and ~DOWN are asserted during the reset operation of the DFPC; and
3. The operation of the circuit when the reference and VCO output frequencies are an octave apart. Since the circuit is symmetrical the test need only be done once for either case where the reference is twice that of the VCO frequency, or vice versa.

Figure 3 shows the test circuit for generating signals of different frequencies. A ring oscillator circuit contained within the device generates a base frequency that is used as the reference signal into the DFPC. The reference signal is fed into a frequency divider and the output of the divider drives the VCO input into the DFPC.

Figure 3 also shows the circuit for generating signals with a phase difference. The same ring oscillator used in the previous test circuit is used to generate a base frequency that provides the reference signal. This reference is also used as the input into a programmable phase shifter. The phase shifter is controlled using the standard device down-loading software normally used for loading the device bit stream from a personal computer. Phase adjustments are made in increments of 100 ps.

This phase shifted signal, which has the same base frequency as the reference, is used as the VCO input for the DFPC.

When compared to a stable reference frequency, initial tests show that at typical operating conditions, the DFPC has a common mode pulse width, t_{CMPW}, of approximately 10.5 ns. Figure 4 shows the time difference between a level change in ~UP and ~DOWN, t_{DPW}, is exactly equal to the phase shift of the two input signals, t_{DIP}. The symmetry of the circuit layout in the AT6000 device enables the difference between t_{DPW} and t_{DIP} to be less than 500 ps.

The performance and utilization statistics are given in Table 1. The DFPC implementation is available in schematic and layout form.

Figure 2. DFPC Implementation

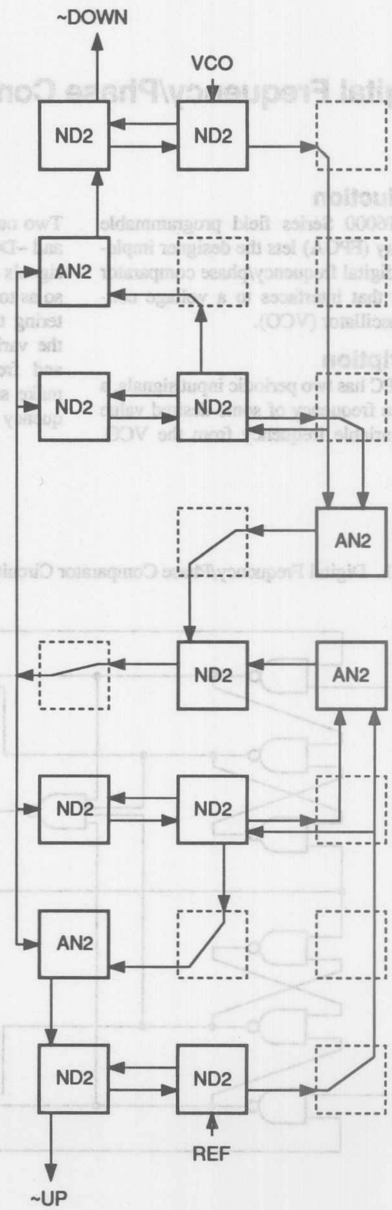
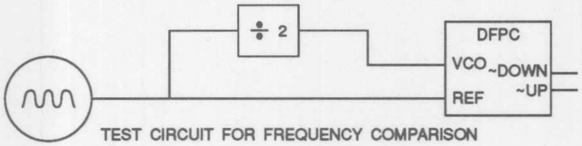


Figure 3. DFPC Test Circuits



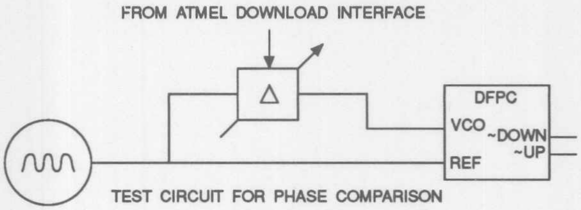


Figure 4. DFPC Timing Diagram

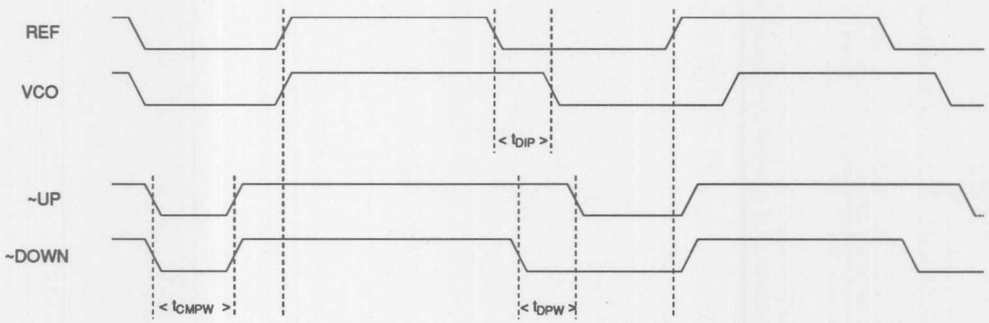


Table 1. 16 x 8 FIFO

	Cell Count(1)	Minimum Bounding Box (X x Y)	t_{CMPW}(2)	t_{DPW} - t_{DIP}(2)
DFPC	24	3 x 8	10.5 ns	0.5 ns

Notes:
1. Includes cells used as wires.
2. Typical Operating Conditions: 25°C, 5.0 V, AT6000-4.

Field Programmable Gate Array

Application Note

Configuration Compression Algorithm

Introduction

AT6000 Series FPGAs are SRAM-based and can be reconfigured to perform different applications in a system. Formulas show how the act of reconfiguration affects system performance. A proprietary compression algorithm reduces reconfiguration time and improves system performance. This algorithm is incorporated into the bit stream generation software provided in the Integrated Development System.

Description

Two factors determine configuration time—the frequency of the configuration clock and the configuration mode used.

The configuration clock, CCLK, regulates the loading of data. The higher the clock frequency, the faster data is loaded. Modes 4 and 5 use an internally supplied clock that runs at only 1 MHz. The other modes employ a user-supplied clock. The user-supplied clock can run as fast as 10 MHz.

Of the two kinds of configuration, serial and parallel, parallel configuration is faster. Serial configuration loads one bit per clock cycle, while parallel configuration loads eight bits per clock cycle. As a result, a serial configuration mode takes eight times longer to load a bit stream. Table 1 gives equations that determine configuration times for the AT6002 and AT6005.

Partial configuration is naturally faster than full configuration. A configuration compression algorithm, supplied with Atmel's development system, filters full-configuration data to produce a partial configuration bit stream. The bit stream produced by the compression algorithm only programs memory that is different from the present configuration. On power-up, for example, each cell in the array is a zero. The compression algorithm can remove all the zeros from the bit stream to be used after power-up—in some cases reducing the size of the bit stream by as much as 50%.

6

Table 1. Configuration Timing Equations

Without Compression	**AT6002**	**AT6005**
Serial	2678 x 8 x 1/Frequency	8077 x 8 x 1/Frequency
Parallel	2678 x 1/Frequency	8077 x 1/Frequency
With Compression	**AT6002**	**AT6005**
Serial	(1 - % Reduction) x 2678 x 8 x 1/Frequency	(1 - % Reduction) x 8077 x 8 x 1/Frequency
Parallel	(1 - % Reduction) x 2678 x 1/Frequency	(1 - % Reduction) x 8077 x 1/Frequency

Modeling Device Power Consumption

Field Programmable Gate Array

Application Note

Introduction

The following provides a simple method for modeling the active and static power consumption of a AT6005 design.

Active Power Consumption

Active power consumption is a function of the distribution of resources in a design and the number of nets switching each second. The distribution of resources is calculated by counting the instances in the design database. The Integrated Development System (IDS) reports this information in the list files generated by programs like placement, routing and bit stream generation. The switching of some nets, like clock signals and flip-flop outputs, is determined by clock frequency and can be tabulated exactly. The switching of other nets, especially combinatorial logic, is input-dependent and not solely determined by the clock. As a result, the activity of these nodes can only be estimated. Combinatorial signals are typically half as active as the clock. Test vectors representative of actual design operation can give a more accurate calculation. Viewlogic's "activity" command calculates the number of active nodes in a design during logic simulation.

The equation for active power consumption is as follows:

$$POWER = Frequency \times (Aa \times Ka \times Na + Ab \times Kb \times Nb + Al \times Kl \times Nl + Ax \times Kx \times Nx + Ac \times Kc \times Nc + Kg \times Ng + Ai \times Ki \times Ni + Ao \times Ko \times No) \times V_{CC}$$

The N coefficients represent the design resources reported by the IDS:

- *Na* number of A-type nets used (individual cell function is not important)
- *Nb* number of B-type nets used (individual cell function is not important)
- *Nl* number of local-bus type nets used
- *Nx* number of express-bus type nets used
- *Nc* number of clock columns used
- *Ng* 1 if global clock is used
- *Ni* number of I/O inputs
- *No* number of I/O outputs with no output load used

The A coefficients represent the estimated activity of combinatorial logic.

The K coefficients represent the weighting factor of each component:

$Ka = 2\ \mu A/MHz$	$Kb = 2\ \mu A/MHz$
$Kl = 4\ \mu A/MHz$	$Kx = 3\ \mu A/MHz$
$Kc = 100\ \mu A/MHz$	$Kg = 200\ \mu A/MHz$
$Ki = 4\ \mu A/MHz$	$Ko = 60\ \mu A/MHz$

The amount of activity possible is based on the number of each cell type used. The AT6005 has the following available:

$Na = 3136$	$Nb = 3136$
$Nl = 1568$	$Ne = 1568$
$Nc = 56$	$Ng = 1$

$Ki = 64$ (84-pin) or 108 (132-pin)

$Ko = 64$ (84-pin) or 108 (132-pin)

If every node were active at 10 MHz, the device would use about 293 mA of current (1466 mW).

A more typical example would be:

$Na = 2000$	$Ab = 2000$
$Al = 1200$	$Ae = 700$
$Ac = 27$	$Ag = 1$
$Ki = 54$	$Ko = 54$

Yielding an active power consumption of:

$$POWER = 2000 \times 0.5 \times 2 + 2000 \times 0.5 \times 2 + 1200 \times 0.5 \times 4 + 300 \times 0.5 \times 3 + 27 \times 1 \times 100 + 1 \times 1 \times 200 + 54 \times 0.5 \times 4 + 54 \times 0.5 \times 60$$

$$= 11.5\ mA/MHz$$

Or, 115 mA at 10 MHz (575 mW at 10 MHz).

6

Quiescent Power Consumption

The AT6005 is a CMOS device. Once programmed, the SRAM used to store the configuration requires no static power. The programmable interconnect points use complementary CMOS pass gates; this insures that all signals eventually reach V_{CC} or GND and dissipate no static power. There are no passive pull-ups on any internal nodes. Unused nets and buses are tied to V_{CC} and GND, and dissipate no power. Tri-states without active drivers dissipate some static power, but this is easily avoided.

Static power dissipation, measured after power-up in modes 1, 2, 3, or 6, but before programming, is 2 mA. After power-up, the device is programmed as a large array of registers with no inputs connected. Modes 4 and 5 generate a clock output signal. The power dissipation of modes 4 and 5 is 2 mA plus the power dissipation of the CCLK output driver, which is a function of the pin's loading capacitance. CCLK is typically 1 MHz.

The primary source of static power dissipation is not the core array, but the SRAM configuration circuitry. It has two blocks which consume static power—a power supply voltage monitor and an internal oscillator. The voltage monitor is used to initiate reboot when V_{CC} is first applied or when V_{CC} goes below a critical voltage. The monitor can not be disabled. The internal oscillator can be turned off by setting the B5 bit in the configuration register. With B5 set, the AT6005 dissipates less than 900 µA static power (Table 1).

Power consumption calculation is performed automatically by the Integrated Design System.

Table 1. AT6005 Static Power Dissipation

		Min	Typ	Max
ICC1	Modes 1, 2, 3, 6 Measured after reboot			2 mA
ICC1	Modes 4, 5 With 50p load on CCLK		5 mA	
ICC2	Modes 1, 2, 3, 4, 5, 6 B5 Bit set with CONN=CSN=VCC		900 µA	

Converting FPGAs and PLDs to Atmel Gate Arrays

FPGA/PLD to Gate Array Conversion

Application Note

Introduction

Atmel is one of the only companies which designs and manufactures field programmable gate arrays (FPGAs), programmable logic devices (PLDs) and high performance gate arrays. In a natural progression, Atmel offers a seamless, direct conversion path for designs implemented on most PLDs and FPGAs to its gate array families. The potential benefits to the system designer of such a capability are fourfold:

- Component cost savings. Atmel's conversion process will convert a single FPGA or PLD into a lower cost gate array that is a pin-for-pin compatible replacement.
- Board space savings. Atmel converts to a true gate array, not a hardwired FPGA/PLD. Multiple FPGAs or PLDs can be converted and consolidated into a single gate array, reducing system component count and providing even more cost savings.
- Enhanced performance. Conversion to a gate array grants the designer access to all of the macrocells and functions contained in the cell library, including higher order logic functions and testability improvement circuitry that cannot be realized on an FPGA or PLD. Gate array routing schemes allow a greater degree of flexibility to optimize timing performance or logic area.
- Reduction in design cycle time. An ASIC design can be prototyped using programmable logic and migrated to a gate array for production without the time and cost of a redesign.

In all cases, Atmel uses the existing FPGA or PLD design data base so that little additional engineering effort is required from the customer. This application note discusses some factors to consider when deciding to convert, describes the conversion process, and details the required information for selected FPGA and PLD products.

ATL Array Organization - 1.0 μ CMOS

Device Number	Raw Gates	Routable Gates	Max Pin Count	Max I/O(1) Pins	Gate(2) Speed
ATL4	4,100	2,600	68	60	375 ps
ATL10	10,000	6,500	124	116	375 ps
ATL20	22,000	12,000	144	136	375 ps
ATL40	40,000	22,000	180	168	375 ps
ATL60	57,000	30,000	224	208	375 ps
ATL75	72,000	38,000	256	236	375 ps
ATL100	95,000	50,000	292	262	375 ps
ATL130	131,000	67,000	338	308	375 ps
ATL160	157,000	80,000	360	320	375 ps

ATL C Fine Pad Pitch Array Organization - (Commercial) 1.0 μ CMOS

Device Number	Raw Gates	Routable Gates	Max Pin Count	Max I/O(1) Pins	Gate(2) Speed
ATL7C	7,000	4,000	100	92	375 ps
ATL10C	10,000	6,000	120	112	375 ps
ATL15C	15,000	8,000	144	136	375 ps
ATL20C	22,000	12,000	160	152	375 ps
ATL35C	35,000	18,000	208	192	375 ps
ATL55C	55,000	29,000	256	236	375 ps
ATL75C	75,000	39,000	304	280	375 ps

Notes: 1. Absolute maximum I/O pins is maximum pin count minus eight. Additional power and ground pins may be required to support simultaneous switching outputs as pin count increases.

2. Nominal two input NAND gate with a fanout of two.

ATL80 Array Organization - 0.8 μ CMOS

Device Number	Raw Gates	Routable Gates	Max Pin Count	Max I/O(1) Pins	Gate(2) Speed
ATL80/15	17,000	10,200	100	92	256 ps
ATL80/25	26,000	15,600	120	112	256 ps
ATL80/40	39,000	23,400	144	136	256 ps
ATL80/50	50,000	30,000	160	152	256 ps
ATL80/75	75,000	45,000	184	176	256 ps
ATL80/95	94,000	60,000	208	192	256 ps
ATL80/150	150,000	75,000	256	236	256 ps
ATL80/220	220,000	110,000	304	280	256 ps
ATL80/280	280,000	140,000	340	310	256 ps
ATL80/350	350,000	175,000	380	350	256 ps
ATL80/450	450,000	225,000	424	384	256 ps
ATL80/600	600,000	300,000	480	440	256 ps

Notes: 1. Absolute maximum I/O pins is maximum pin count minus eight. Additional power and ground pins may be required to support simultaneous switching outputs as pin count increases.

2. Nominal two input NAND gate with a fanout of two.

Programmable Logic vs Gate Array

Programmable logic has enjoyed tremendous popularity and growth over the last several years, primarily because the user saves both time and money. Designers may work with inexpensive design tools which run on inexpensive platforms. Designs can be implemented in hours and modified easily. A designer can implement an ASIC design and evaluate its performance in a system in the same week. This instant feedback allows designers to validate system operation and rectify any errors without additional expense. Programmable logic devices provide an ideal solution for fast prototyping and for low to moderate production volumes.

Gate arrays, however, continue to offer superior performance, higher density, and lower cost-per-gate in volume production when compared to an FPGA or PLD. While the cost of gate array design tools has dropped in recent years and quality third party tools have emerged, they are still more expensive than FPGA or PLD design tools. Also, while gate array prototypes can be delivered in days or weeks, that is still a much longer period than the several hour turnaround of a programmable logic device.

When to Convert

There are four instances when converting from an FPGA or PLD offers the user a direct benefit.

High Volume. If the annual volume for a single, commercial ASIC design exceeds 10,000 units, or if consolidated, commercial designs exceed 2,000 units, converting to a gate array can save money.

Performance. Gate arrays have lower standby current, offer greater speed than an FPGA or PLD, offer a greater selection of buffer types and drive current, and offer a greater selection of higher order logic functions than an FPGA or PLD.

Integration. Converting several FPGAs or PLDs and consolidating the logic into a single gate array uses less board space, reduces component count, and improves the manufacturability and reliability of the system.

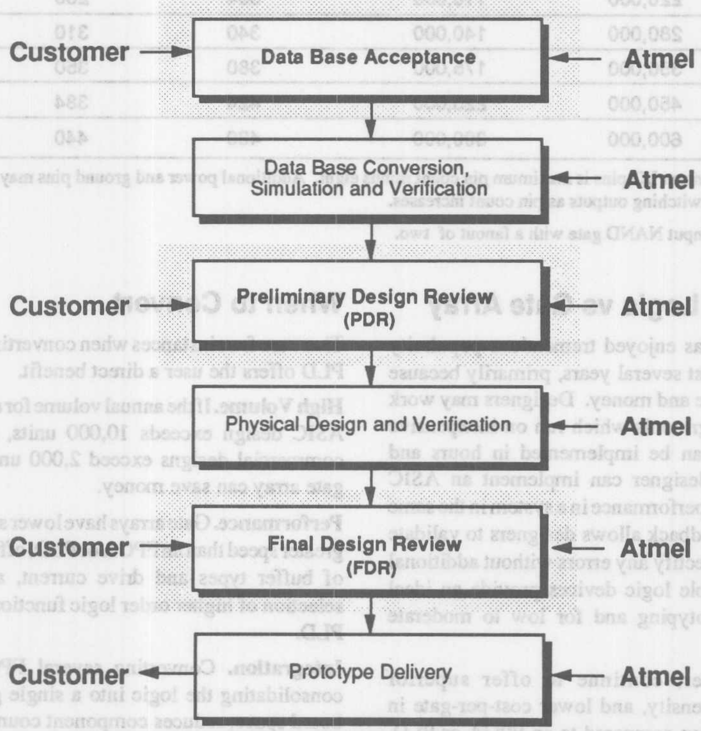
Prototypes. Using the FPGA or PLD for prototypes and then converting to a gate array for production gives the designer the best of both worlds – a fast, accurate design cycle and a lower cost component in production.

Atmel Conversion Process

Summary of the Conversion Process

Atmel's FPGA or PLD to gate array conversion process is intended to minimize the amount of additional engineering support required from the system designer. Figure 1 outlines the conversion process flow. The inputs to this flow will vary depending upon the manufacturer of the FPGA(s) or PLD(s) to be converted. Once the required design data base has been delivered to Atmel and accepted, the design process (Preliminary Design Review thru Prototype Delivery) is the same as that of any other gate array design.

Figure 1. FPGA/PLD to Gate Array Conversion



Required Data Base

Whether converting from FPGA or PLD, the designer must bear in mind that the result of the conversion process is a gate array, and that the array may consolidate the logic contained on several devices onto one chip. It remains the responsibility of the designer to completely define the requirements of the gate array device. The designer will supply data bases that are unique to the specific choice of FPGA or PLD as a part of the conversion process, as well as information that is independent of FPGA or PLD choice. The exact files required from the CAD system used to develop the FPGA/PLD are listed in the following section. In addition, the following data is also required:

Simulation and Test Vector Format

- All vectors are in the same five group format (Input, Output, Tri-State, BiDirectional and Enable), and have a stated purpose
- Outputs are sampled once per clock cycle at the 75% cycle point
- Test vectors must include a 1-MHz set for probe and an "at-speed" set for final test

Specifications

- Operating conditions of temperature and voltage
- System loading requirements by pin
- Operating clock speed and number of clocks
- I/O definition including pin out and enable for Tri-State, and BiDirectional buffers
- Definition of critical paths
- Definition of asynschronous behavior

Documentation

- Full hierarchical schematics
- Clocktree and reset diagram
- Timing diagrams showing relationship of clocks to data applied and valid outputs

The specific data base that is required will depend upon the manufacturer and platform for the FPGA or PLD design. This data will also be dependent on whether one FPGA or PLD is being converted into a single gate array, whether multiple FPGAs or PLDs are being converted and consolidated, and what specific performance is required. The approach that Atmel takes to the conversion process will depend upon the response to the last item.

Atmel's gate array families allow the designer flexibility in I/O placement. Only the eight corner pads on the die are dedicated power or ground. All other pads on the die are fully programmable for input, output, BiDirectional, Tri-State, power or ground.

Converting FPGAs

FPGAs are similar to a true gate array in both design and physical implementation, making the conversion process very simple. Most FPGAs use only a few logic blocks which are a subset of most gate array cell libraries. This allows the FPGA netlist to be mapped directly into gate array primitive cells, simulated, validated, and routed in the same manner in which a gate array is designed. Any optimization that is required to meet performance specifications can also be performed.

Actel FPGAs - Required Data Base

There are several third party design platforms and tools that support programming of an Actel FPGA, including Mentor, Cadence/Concept, and Viewlogic. The required inputs from each of these platforms are listed below.

Mentor Files (via mifexpand.com script)

- .MIF (Mentor Interface File - Netlist)
- .LOG (Simulation Log File)
- .LIST (Simulation Listing File)
- .FORCE (Simulation Force File)

Cadence/Concept Files

- EDIF Netlist (via Synlink)
- ASCII File of Simulation Vectors

Viewlogic Files

- .ADL (Flattened Netlist)
- .CMD (Simulator Command File)
- .EDN (EDIF Netlist)
- .PIN (Final Pin List)
- .VSM (Viewlogic Simulator Netlist)
- .SCH (Schematics)
- .SYM (Symbols)
- .WIR (Wire List)
- ASCII File of Simulation Vectors

Additional files that used to satisfy some of the other data requirements:

Actel Files

- .CRT (Criticality File)
- .DEL (Actual Delay File)
- .FUS (Fuse File)
- .IPF (Initial Placement File)
- .DEF (Parameters File)

Table 1 lists the recommended Atmel gate array for the Act 1, Act 2 and Act 3 families of Actel FPGA.

Table 1. Actel FPGA/Atmel Gate Array Cross Reference

Actel FPGA		Equivalent Usable Gates	I/O Pins	Target Atmel[1] Gate Array
Act 1	1010	1,200	57	ATL4
	1020	2,000	69	ATL4 or 7C
Act 2	1225	2,500	83	ATL4 or 7C
	1240	4,000	104	ATL7C or 10C
	1280	8,000	140	ATL15C, or ATL20C
Act 3	1415	1,500	80	ATL7C or ATL80/15
	1425	2,500	100	ATL10C or ATL80/25
	1440	4,000	140	ATL20C, ATL80/40 or ATL80/50
	1460	6,000	168	ATL35C or ATL80/50
	14100	10,000	228	ATL35C, ATL55C, ATL80/95 or ATL80/150

Note: 1. Target array dependent on number of pins used, and pinout.

Xilinx FPGAs - Required Data Base

There are several third party design platforms and tools that support programming of a Xilinx FPGA, including Mentor, Cadence/Concept, and Viewlogic. The required inputs from each of these platforms are listed below.

Mentor Files (via mifexpand.com script)
- .MIF (Mentor Interface File - Netlist)
- .LOG (Simulation Log File)
- .LIST (Simulation Listing File)
- .FORCE (Simulation Force File)

Cadence/Concept Files
- EDIF Netlist
- ASCII File of Simulation Vectors

Viewlogic Files
- .ADL (Flattened Netlist)
- .CMD (Simulator Command File)
- .EDN (EDIF Netlist)
- .PIN (Final Pin List)
- .VSM (Viewlogic Simulator Netlist)
- .SCH (Schematics)
- .SYM (Symbols)
- .WIR (Wire List)
- ASCII File of Simulation Vectors

Additional files that are used to satisfy some of the other data requirements:

Xilinx Files
- .LCA
- .XNF (Xilinx Netlist)

Table 2 lists the recommended Atmel gate array for a variety of Xilinx FPGAs.

Table 2. Xilinx FPGA/Atmel Gate Array Cross Reference

Xilinx FPGA	Equivalent Usable Gates	I/O Pins	Target Atmel[1] Gate Array
2064	1,200	58	ATL4
2018	1,800	74	ATL7C,ATL4
3020	2,000	64	ATL7C, ATL4
3030	3,000	80	ATL7C
3042	4,200	96	ATL7C, ATL10C
3064	6,400	120	ATL10,10C
3090	9,000	144	ATL20,20C,35C
4002	2,000	64	ATL4,7C or ATL80/15
4003	3,000	80	ATL7C or ATL80/15
4004	4,000	96	ATL10 or ATL80/15
4005	5,000	112	ATL10C or ATL80/25
4006	6,000	128	ATL15C,10 or ATL80/40
4008	8,000	144	ATL20,20C or ATL80/50
4010	10,000	160	ATL20,20C or ATL80/50
4013	13,000	192	ATL35C,40 or ATL80/95
4016	16,000	208	ATL35C or ATL80/150
4020	20,000	240	ATL55C or ATL80/150

Note: 1. Target array dependent on number of pins used, and pinout.

Converting PLDs

Two different methods can be used when converting from PLDs to gate arrays, each of which is intended to provide an optimal solution to a specific concern. These methods are deterministic and timing matching. Each method involves trading gate array utilization for matching the timing of the original design, as can be seen in Table 3.

Deterministic Conversion

PLDs have a uniform, deterministic architecture. Every signal within a PLD traverses a constant length path and avoids race conditions. If the converted design is to be a drop-in replacement, then it must meet the same specifications of the original PLD design, including minimum and maximum signal arrival times, set-up and hold times.

These system requirements become particularly critical when the PLD is driving a chip with a positive hold time. As Figure 2 demonstrates, a reduction in the clock-to-output time will cause a hold time failure within the system.

Using a deterministic approach, the gate array logic is implemented using blocks similar in structure to those used in the PLD, i.e., product terms and sum terms. This methodology eliminates the possibility of introducing the timing problem described above and is moderately efficient in terms of gate utilization and timing matching.

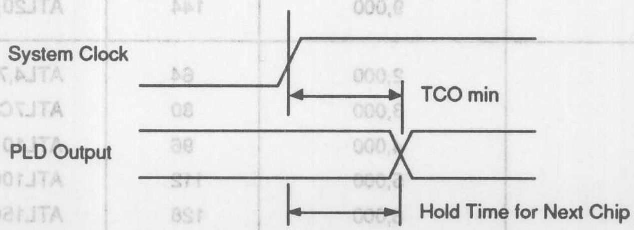
Timing Matching Conversion

If a PLD design is completely synchronous, timing optimization techniques can be used to find the minimum propagation delay for each path. This information can then be used to adjust delay to paths until the maximum signal arrival time of the path, as implemented in the gate array, matches that of the PLD. As the name of this approach implies, this method offers the best timing match between gate array and PLD, at the expense of additional buffers to adjust specific path timing.

Table 3. PLD to Gate Array Conversion Methodologies

Conversion Methodology	Gate Utilization	Timing Match	Comments
Deterministic	Moderate	Moderate	Eliminates internal timing concerns.
Timing Matching	Lowest	High	Eliminates both internal and system timing concerns.

Figure 2. System Timing Concerns (Positive Hold Times)



Atmel FPGAs/PLDs - Required Data Base

The specific file requirements for converting from an Atmel FPGA or PLD are quite straightforward. Table 4 is a cross reference for Atmel PLD/FPGA and gate arrays.

JEDEC Files (PLD)

- ABEL
- CUPL
- LOGIC

Viewlogic File (PLD, FPGA)

- .GDF (Graphic File)
- .TDF (Text File)
- .HIF, .FIT (Fitter File)
- POF (Programmer Object File)
- .SNF (Simulation Netlist File)
- .VEC (Simulation Vector File)

IDS File (FPGA)

- .CDB

Table 4. Atmel PLD and FPGA/Atmel Gate Array Cross Reference

		Equivalent Usable Gates	I/O Pins	Target Atmel[1] Gate Array
PLD	AT22V10	300	10	ATL4
	AT18V8	250	8	ATL4
	ATV750	750	10	ATL4
	ATV2500	2,500	24	ATL4, ATL7C
	AT3000	3,000	56	ATL7C
	ATV5000	5,000	52	ATL10C
FPGA	AT6002	2,000	96	ATL7C, ATL80/15
	AT6003	3,000	108	ATL10C, ATL80/25
	AT6005	5,000	108	ATL10C, ATL80/25
	ATL6010	10,000	160	ATL35C, ATL80/75

Note: 1. Target array dependent on number of pins used, and pinout.

As discussed earlier, the type of conversion approach that is selected will determine how many of a particular PLD can fit onto an array. Atmel's ATL4 array, featuring up to 2,600 routable gates and 68 die pads, is used as the target array for the following example.

For a deterministic conversion, the percentage utilization is given by the formula:

$$\% \textit{ utilization} = (0.15x + 0.008y + 1.5z)$$

where x is the number of product terms, y the number of pins, and z the number of registers.

For a timing matching conversion, the percentage utilization is given by:

$$\% \textit{ utilization} = (0.35x + 0.08y + 1.5z)$$

Figure 3 shows a range for the number of PLD designs which will fit onto an ATL4, for selected members of Atmel's ATV family and for both deterministic and timing matching approaches. There will often be a wide range of PLD utilization within a given system configuration. The value shown for the "median" will provide a good benchmark when considering converting multiple PLDs into a gate array. The maximum number shown is typically a function of the number of I/O pins required.

Altera FPGA/PLD - Data Base Required

Converting from Altera FPGA/PLD into an Atmel gate array follows much the same process as converting from an Atmel FPGA/PLD. The specific file requirements for converting from an Altera FPGA/PLD are the Abel, Cupl or other JEDEC based development system files plus the ASCII file of the vectors. Table 5 lists the recommended Atmel gate array for both the Max Series and Classic Series of Altera FPGA/PLD. When conversion from a Max series device is desired, there are other files, generated from the MAX+PLUS design system, that can be provided to satisfy some of the data requirements.

Using Altera inputs, an FPGA/EDIF 200 netlist with actual delay data can be easily extracted. This netlist comes from the MAX+PLUS design system and incorporates the delay data from the fitter/assembler. These files are:

MAX+PLUS Files

Archive File (Menu: File/Project/Archive)

Should contain these files as a minimum:

- .GDF (Graphic File)
- .TDF (Text File)
- .HIF, .FIT (Fitter File)
- .POF (Programmer Object File)
- .SNF (Simulation Netlist File)
- .VEC (Simulation Vector File)

Figure 3. Number of Specific PLD Designs which will fit in an ATL4 (Min, Median, Max)

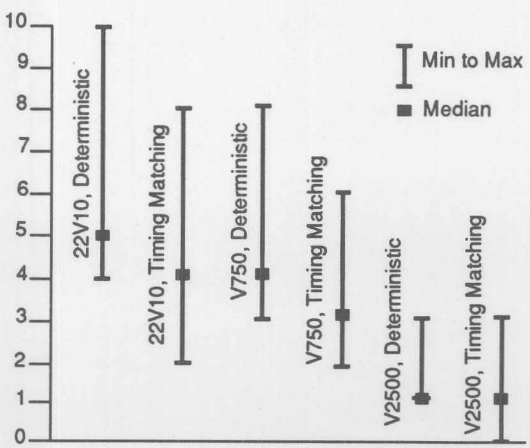


Table 5. Altera FPGA/PLD/Atmel Gate Array Cross Reference

Altera PLD		Equivalent Usable Gates	I/O Pins	Target Atmel[(1)] Gate Array
Classic Series	320/30	150	18	ATL4
	610/30/10A	300	20	ATL4
	910	450	36	ATL4
	1810	900	54	ATL4
Max 5000	5016	300	16	ATL4
	5032	600	24	ATL4
	5064	1,200	36	ATL4
	5128	2,400	60	ATL4
	5130	2,400	84	ATL4, 7C
	5192	3,500	72	ATL7C
Max 7000	7096	2,000	76	ATL4, ATL7C
	7128	2,500	100	ATL7C, ATL10C
	7160	3,200	104	ATL10C
	7192	3,800	124	ATL10
	7256	5,000	164	ATL20C, ATL35C
	7320	6,400	164	ATL20C, ATL35C
	7384	7,500	196	ATL20C, ATL35C
	7512	10,000	196	ATL20C, ATL35C
	7768	15,000	224	ATL35C, ATL55C
	71024	20,000	260	ATL35C, 55C, 75C
Altera FPGA		**Equivalent Usable Gates**	**I/O Pins**	**Target Atmel Gate Array**
Flex 8000	8452	4,000	116	ATL10C, ATL15C or ATL80/25
	8820	8,000	148	ATL20C or ATL80/60
	81188	12,000	180	ATL35C or ATL80/90
	81720	18,000	150	ATL55C or ATL80/150
	82252	24,000	236	ATL75C or ATL80/150

Note: 1. Target array dependent on number of pins used, and pinout.

Gate Array Implementation

After data base acceptance, the design data base is converted into an equivalent netlist of primitive cells from Atmel's gate array library. The vectors from the original FPGA or PLD design are also converted and are used as functional simulation vectors to validate the gate array netlist. As these vectors are used to perform any timing simulation and form the core of the gate array tester program, it is vitally important that an accurate and complete vector set is provided.

After the FPGA or PLD data bases have been converted and validated, any additional circuitry, such as memory blocks, testability improvement elements, or higher order logic functions, can be incorporated into the netlist. Any optimization that is necessary to match timing or to improve performance can be performed at this point as well. At this point, boundary and internal scan can be added and ATPG vectors generated. A Preliminary Design Review is then held with the customer to review and to approve the results of the design conversion.

Preliminary Design Review (PDR)

The following items are reviewed at the PDR:

- Confirm v3 Netlist Checker and tvc Test Vector Checker files correct
- I/O buffer listing and bonding diagram
- Preliminary testability compiler report
- Route clock tree and analysis of worst case and best case delay
- Verilog simulation at-speed
  - -nominal, worst case, best case (with no timing violations)
- Review critical path information (t_{SU}, t_{HOLD}, t_{PD})
  - -Verilog or Veritime estimates
- I/O electrical specifications
- Electromigration calculation

Final Design Review (FDR)

Beyond this point, the design process follows that of a traditionally designed gate array. The cells are placed and routed, a post-route simulation is performed, and checks are performed to verify conformance with electrical and design rules, and to confirm the logic versus schematic is correct. An FDR is held with the customer to review and approve the post route data, and to authorize mask making and prototype fabrication.

The FDR is the last joint review between Atmel and the customer before committing to prototypes. Prior to this meeting, both Atmel and the customer will have reviewed the post-route Verilog-XL simulation incorporating the back annotation data. The customer will receive back annotation data for complete post-route simulation on their CAE systems. Atmel guarantees silicon performance equal to or better than that predicted by the post-route Verilog-XL simulations. The items to be reviewed at FDR are as follows:

- Updates of cell mapping and timing (if any)
- Post-route netlist check v3
  - -post-route netlist changes
- Post-route timing simulation to specification
  - -review clock timing
  - -at speed
  - -clock skew (if required)
  - -listing of timing warnings with explanation
- Static path analysis (as specified)
- Electromigration calculation
- Bonding diagrams and pin list
  - -bond pad plot
- LVS/DRC/ERC

Prototype Delivery

Atmel will deliver 10 prototypes in ceramic packages to the customer. The units are to verify the functionality and electrical performance of the gate array.

Synthesis from a Hardware Description Language (HDL)

There has been an increase in the use of HDLs to design FPGAs and PLDs as more of the design platforms offer this capability. Two of the most popular such languages are VHDL and Verilog-HDL. Using a logic synthesis technique, the behavioral level description of an FPGA or PLD can be mapped into a functionally equivalent gate array netlist. Both hardware description languages are supported by the Synopsys Design Compiler. This FPGA/PLD to gate array conversion methodology requires the least amount of data conversion and allows the flexibility to incorporate such features as memory, testability, or higher order logic functions into the gate array. This technique is also effective when the need to consolidate several FPGA or PLD designs into one gate array exists. Synthesis from an HDL offers the most efficient utilization of the gate array, at the expense of timing matching. Should the user require them, VHDL descriptions of the converted FPGAs or PLDs, as well as the gate array implementation, can be provided by exporting the netlists through Synopsys.

Testability Improvement and Automatic Test Pattern Generation

The incorporation of testability improvement circuitry into an ASIC design becomes more important as the density of the design increases. The same can be said for conversion and consolidation of large numbers of dense FPGAs or PLDs into a gate array. The insertion of scan paths within an ASIC and testing via ATPG can provide an easy means of screening manufacturing-related defects during testing, with a relatively small silicon usage penalty. Using ATPG is only a supplement to functional test vectors, not a replacement.

The process consists of replacing existing flip-flops with scan flip-flops and connecting them up to form scan chains. An input pin and output pin must be identified for each scan chain. In general, scan chains should not exceed 64 flip-flops in length. Thus, for a design with 600 flip-flops, 10 input pins and their corresponding output pins must be identified. Existing pins may be multiplexed for this use if the design is pin limited. Additional pins are required for the Test Enable (TE) signal and a Test Mode (TM) signal. The TE pin is used to control the flip-flops, placing them in either normal mode or scan mode.

The TM pin is required to bypass violations of testability guidelines, an example of which would be gated clocks. During testing, all flip-flops in the scan chains must toggle on the same clock. If gated clocks exist in the design, logic must be designed so that it bypasses this gating when Test Mode is active. Since Test Mode is active only during ATPG test, the basic function of the design is unaffected.

Table 6 outlines other testability rules and suggested workarounds utilizing the Test Mode signal. When all test guidelines are followed, testability insertion and vector generation are easily accomplished. Past experience has shown extremely high fault coverage (up to 99%) with small ATPG vector sets. If these rules are not followed closely, incorporating scan and ATPG can require several weeks. It is highly recommended that the FPGA be designed using the rules in Table 6 if the customer intends to someday convert to a gate array and use scan/ATPG.

6

Table 6. Synopsys Test Compiler Guidelines

Testability Rule	Effects of Infraction	Workaround
Synchronous Design - No cross coupled gates - No unregistered feedback	Associated logic untestable	Break feedback path with test mode
Single Edge Clocking	Clocked device not allowed in scan chain - reduced fault	In test mode, create single edge clocking with inverters and MUXs
No Clock Gating	Clocked device not allowed in scan chain - reduced fault coverage	Use data disable flip-flops instead of clock enables, disable gating in test mode
No Latches	Not allowed in scan chain, reduced fault coverage	Use alternate test methods, force latches to transparent mode with test mode
Single External Reset - No asynchronous resets or presets generated on chip - No combinational logic in reset path	Not allowed in scan chain, reduced fault coverage	Reset OR'd with test mode
No Internal Tristate Buses	Reduced fault coverage, possible tristate contention during scan test	Use MUXs or AOI gates, insert gating of controls to prevent contention
No Direct Q to D Connections	Dynamic Hazard	

Translation Of Existing ASIC Designs

ASIC Design Translation

Application Note

Introduction

It has only been in the last few years that designers and users of application specific integrated circuits (ASIC) have been able to obtain additional sources for these types of integrated circuits. The introduction of design synthesis software by CAD / CAE companies has made the task of converting from one ASIC vendor's library into another's, a feasible task even for the most dense designs.

The user of an application specific integrated circuit who desires the flexibility and security offered by having multiple sources for what are often key system components, or the user who requires an improvement in performance offered by advanced process technologies, now have an easy path to satisfy their needs. That path is Atmel's Design Translation ASIC design flow, shown in Figure 1. The Design Translation flow highlights the major steps that are taken in converting a netlist into Atmel's gate array cells, verifying the translation, performing the layout and realizing the desired circuit performance, and fabricating and testing the resulting silicon product.

This application note describes the types of data required from the ASIC user and the process steps followed by Atmel to successfully translate an existing ASIC design, and presents the results of two translation efforts. The first was an effort where the Atmel device performance was required to match that of the original ASIC. The second was a translation where the improved performance of Atmel's device was required. The process has been proven through successful translation of designs from such vendors as LSI Logic, NEC, Fujitsu, and Oki, into our 1.0 μ ATL series gate array family.

The Process

Simply stated, the Design Translation process maps cells from the original design into cells contained in Atmel's cell library. These cells may be equivalent primitives or may be soft macros which include several primitives. The choice of Atmel cell will depend upon the required performance objective, and, in some instances, hard macros may be created to replace soft macros to achieve performance goals for the design.

Once the mapping is complete, the process follows Atmel's normal ASIC design flow, including cell placement and routing, resimulation using Atmel's "golden" simulator, comparison of predicted versus desired performance and, after approval of the design by the user, PG tape out and prototype fabrication.

6

Data Base Required

ASIC design can be accomplished on a variety of platforms, and with a variety of software tools - some open, some proprietary. Designs are completed using generic and/or vendor-specific library cells as well. This level of flexibility available to the ASIC designer does not hinder the translation effort. Most design tools are capable of providing a netlist in EDIF (EDIF 2.0.0) format. Other netlist formats which are acceptable to Atmel are listed below:

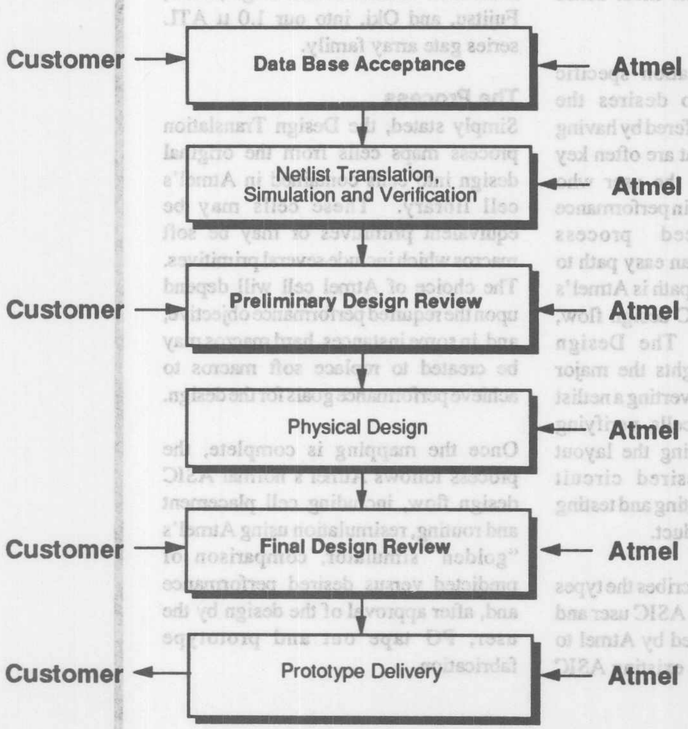
- Cadence™ - Verilog-XL™, EDIF 2.0.0
- DAZIX™ - Tegas™, EDIF 2.0.0
- LSI Logic - NDL™
- Mentor™ - MIF™
- Racal-Redac™ - EDIF 2.0.0
- Valid™ - EDIF 2.0.0
- Viewlogic™ - EDIF 2.0.0

In addition to the netlist for the original design, several other pieces of information are needed to successfully translate the design. Also required are:

- A description of the original design library
- Functional test vectors, print-on-change format
- Device specifications
- Identification of critical and/or asynchronous paths
- "As-routed" delay data from the original design

To make an assessment of whether the desired performance match or improvement has been achieved, an understanding of the starting point must be reached. The description of the original cell library, with its functional and timing information for each cell, is also essential to the definition of the starting point.

Figure 1. Design Translation Flow



If sample parts of the original design can be provided, performance data on the actual silicon can be used to help establish a baseline.

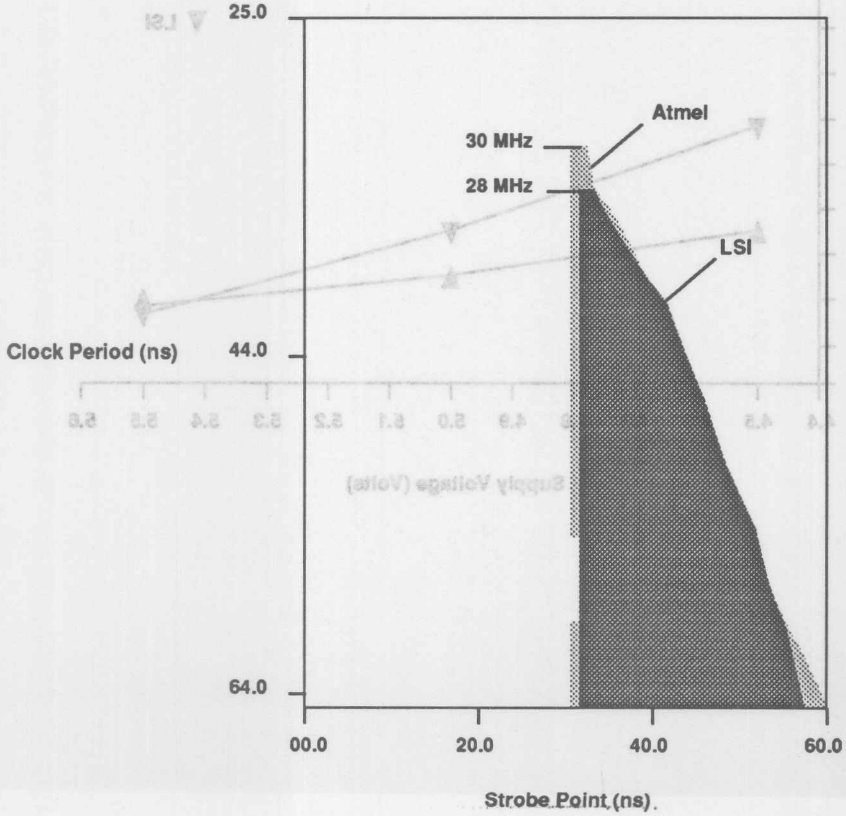
The functional vector set (in ASCII or TSSI, print-on-change format) serves two purposes. The functional vector set, when converted into tester-specific format and used in conjunction with the sample devices, provides a mechanism for establishing detailed performance attributes of the original design. These attributes, such as maximum frequency, path timing performance, and buffer characteristics to name a few, provide the base for cell and buffer selection to match or improve the design. Individual performance attributes can also be used as input to a waveform comparison tool. This tool, using the actual data and the functional test vectors, now converted to simulator format, permits the Atmel designer to determine when and where timing mismatches occur and adjust the netlist accordingly.

The device specification provides key information concerning the required device pin-out, system loading for each pin, the desired performance, and the range of operating conditions.

Performance Matching

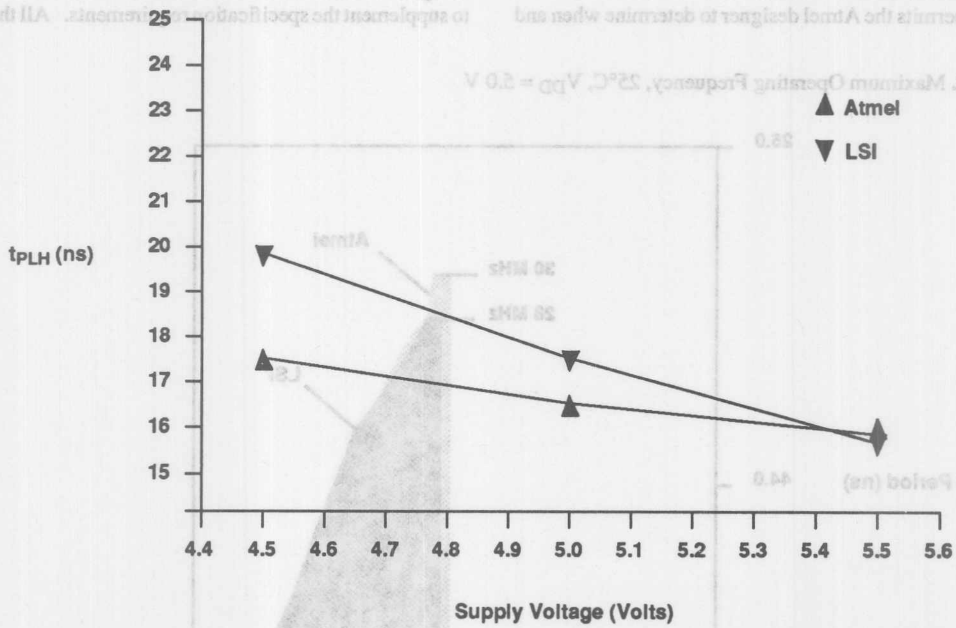
Our first example of ASIC translation presents the results of work performed for a military application, where interchangeability with the original designs was required. The original design was an LSI Logic 10K series gate array of approximately 5,000 gates. The design was asynchronous and had multiple clocks. Samples representing the original designs were available at the outset and were characterized to supplement the specification requirements. All the data

Figure 2. Maximum Operating Frequency, 25°C, V_{DD} = 5.0 V



presented is a direct comparison of LSI Logic and Atmel silicon. Figure 2, depicting maximum operating frequency for constant temperature and voltage, shows how closely the performance can be matched. Figures 3 and 4 depict the average performance for nine critical paths, for low-to-high transitions, both as a function of supply voltage and of temperature. And finally, Figures 5 and 6 depict rise and fall time of bidirectional buffers. The performance match is extremely close.

Figure 3. Atmel vs LSI Package Test Results for 9 Critical Paths, t_{PLH}, +25°C



The customer performed extensive tester-based characterization and qualification of the Atmel device to insure that it was pin-for-pin compatible, drop-in replacement of the original LSI Logic part. The parts were then assembled onto boards and tested again. The Atmel and LSI parts were interchanged and mixed and matched on boards. The complete system evaluation was performed and in all tests the Atmel parts proved to be equal or superior to the LSI gate array.

Figure 4. Atmel vs LSI Package Test Results for 9 Critical Paths, t_{PLH}, $V_{DD} = 5.0$ V

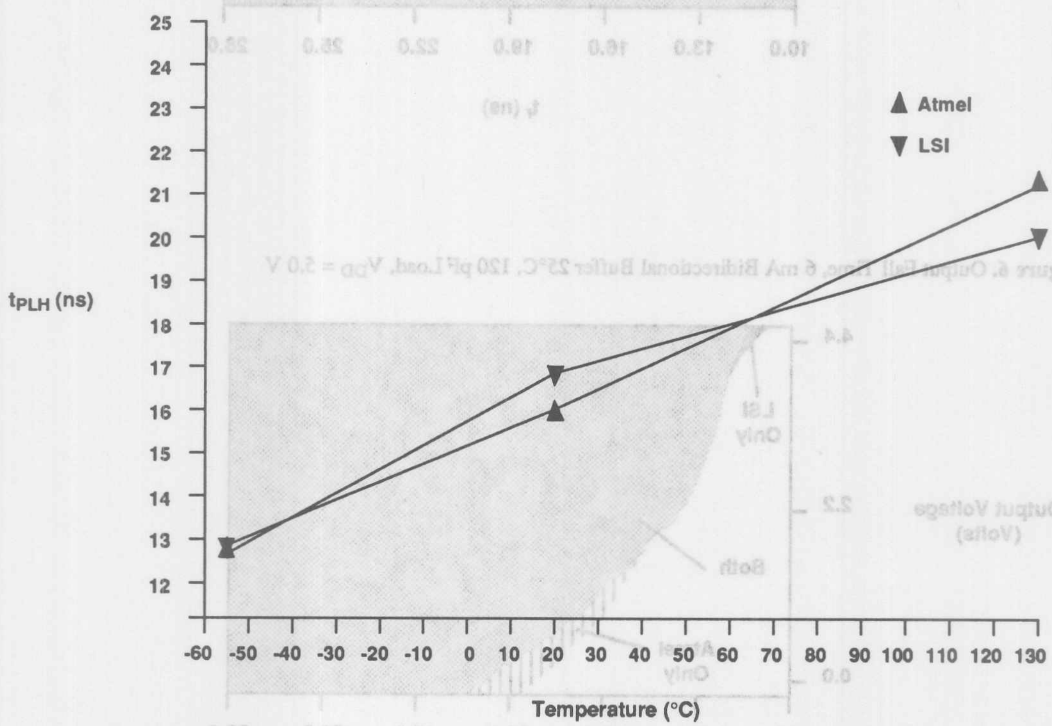


Figure 5. Output Rise Time, 8 mA Bidirectional Buffer, 25°C, 120 pF Load, V_{DD} = 5.0 V

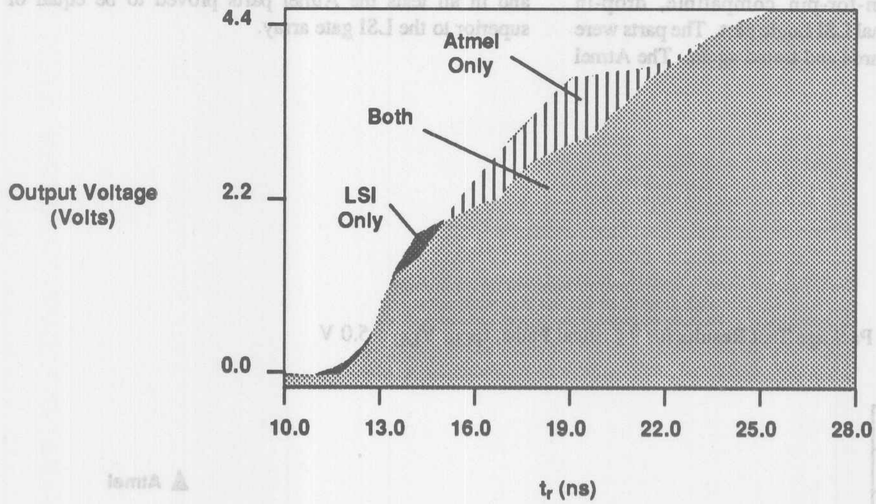
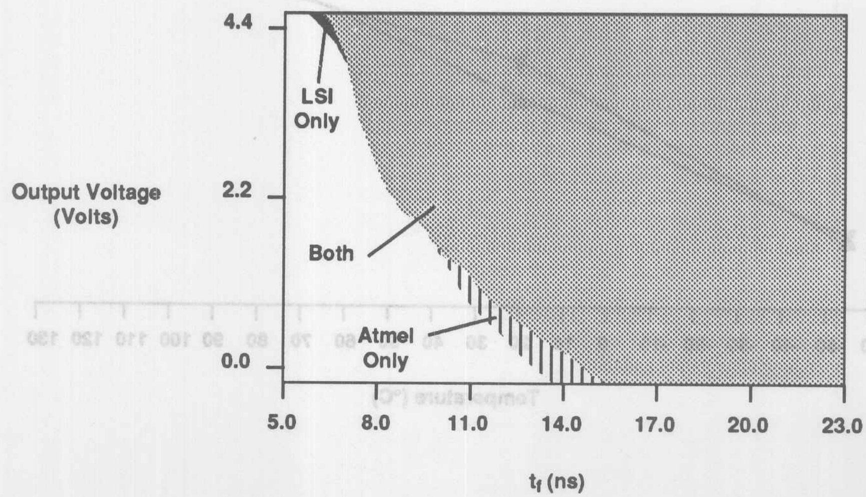


Figure 6. Output Fall Time, 6 mA Bidirectional Buffer 25°C, 120 pF Load, V_{DD} = 5.0 V



Performance Improvement

The second example of ASIC translation presents the results of work performed for a commercial application. This design required approximately 8,000 gates, was completely synchronous, and was operating at 25 MHz. The customer desired an improvement in performance to 33 MHz. To achieve this speed, Atmel compared the performance of its cells to those of the original design, as samples were not available. This evaluation indicated that a 30 to 35 percent speed improvement could be realized over the existing design. Atmel also employed higher drive cells where appropriate to further enhance performance. Extensive board level testing, performed by the customer, confirmed that the Atmel implementation exceeded the 33 MHz design goal over the rated voltage and temperature ranges.

Atmel's CMOS Gate Array Design Manual provides more detailed information about the gate arrays, design methodologies, and individual cell timing, and should be used as a reference for evaluating ASIC performance.

ATMEL

Contents

Standard Military Drawing PLD Offering

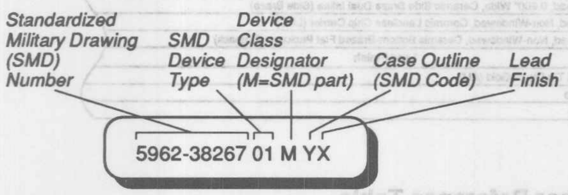
Introduction to the SMD Product Listing

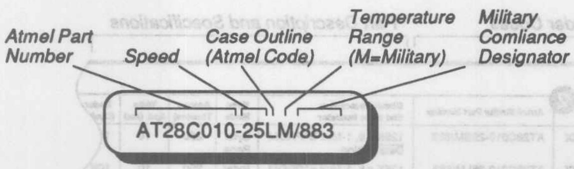
Each Standardized Military Drawing (SMD) part number that Atmel supplies corresponds to an Atmel /883 part number. SMD products are compliant to MIL-STD-883, paragraph 1.2.1 and to the requirements of the applicable standardized military drawing. The tables in this section list the currently approved Atmel SMD parts by Atmel part number (Table 1) and by SMD part number (Table 2). They define and cross reference the Atmel /883 part number with the SMD part number for your ordering convenience.

Figure 1 (below) shows how an Atmel SMD order number defines a part, compared to the components of the Atmel similar part number.

Please note that some SMD part numbers contain the letter "M" between the device type and the case outline designator. The "M" is part of the one part-one part number system, set up by DESC. It is a device class designator which indicates the part is an SMD part number as opposed to being a JAN part number.

Figure 1. Components of an SMD number (top) compared to the Atmel similar part number (bottom).





7

How to Use the Atmel Part Type Reference Table

The organization of Table 1 enables the purchaser to order a standardized military part by using the Atmel generic part type to locate the correct SMD drawing number. The SMD part number is the order number for SMD devices. The Atmel generic part type, which begins with the prefix "AT," heads Table 1. There are four sections in this table (see the sample table below):

SMD Options

The first section lists the SMD options available at Atmel. It includes the industry generic part type, the SMD drawing number, and the SMD device type. The first section also lists the SMD case outline options, the lead finish options, the circuit description, and the access time that correspond to that device type.

Order Code Cross-Reference

The next section of the table cross-references each optionally complete SMD part number (see ❶ below) with the Atmel similar /883 part number (see ❷ below).

Case Outline Legend

The third section gives the SMD case outline options available for the device.

Lead Finish Legend

The last section lists the SMD lead finish options available for the device.

AT28C010

SMD Options with Part Description and Specifications

Generic Number	Standard Military Drawing Number				Description				
28C010	Drawing Number	Device Type	Case Outline	Lead Finish	Circuit Description End Write Indicator	Write Mode	Access Time(ns)	Write Spd(ms)	Endurance (Cycles)
	5962-38267	01	X, Y, Z	X, A, C	128K x 8, 1-Mbit E^2PROM Data Polling	Byte/ Page	250	10	10K
	5962-38267	03	X, Y, Z	X, A, C	128K x 8, 1-Mbit E^2PROM $\overline{\text{Data}}$ Polling	Byte/ Page	200	10	10

Order Code Cross-Reference

Atmel Cage No. 1FN41	Example: Atmel Order Number	Atmel Similar Part Number
	❶ 5962-38267 01M XX	❷ AT28C010-25BM/883
	5962-38267 01M YX	AT28C010-25LM/883
	5962-38267 01M ZX	AT28C010-25FM/883
	5962-38267 03M XX	AT28C010-20BM/883
	5962-38267 03M YX	AT28C010-20LM/883
	5962-38267 03M ZX	AT28C010-20FM/883

Case Outline Legend

Case Outline	
X	32B, 32 Lead, 0.600" Wide, Ceramic Side Braze Dual Inline (Side Braze)
Y	44L, 44 Pad, Non-Windowed, Ceramic Leadless Chip Carrier (LCC)
Z	32F, 32 Lead, Non-Windowed, Ceramic Bottom-Brazed Flat Package (Flatpack)

Lead Finish Legend

Lead Finish	
X	Allows Hot Tin Dip or Gold (AU)
A	Hot Tin Dip
C	Gold (AU)

How to Use the Atmel SMD Number Reference Table

Table 2 allows quick reference to the Atmel /883 similar part number when the purchaser knows the SMD part number. The head for Table 2 is the SMD drawing number (see the example table below). This table contains a cross-reference between the optionally complete SMD drawing number (see ❶ below) and the corresponding Atmel similar /883 part number (see ❷ below). It also includes the circuit description and access time for that part number.

Order Codes — *Part Description and Specifications*

5962-38267

Order Code Cross-Reference

❶ Atmel Order Number	❷ Atmel Similar Part Number	Circuit Description End Write Indicator	Write Mode	Access Time(ns)	Write Spd. (ms)	Endur (Cyc
5962-38267 01M XX	AT28C010-25BM/883	128K x 8, 1-Mbit E^2PROM $\overline{\text{Data}}$ Polling	Byte/ Page	250	10	1
5962-38267 01M YX	AT28C010-25LM/883	128K x 8, 1-Mbit E^2PROM $\overline{\text{Data}}$ Polling	Byte/ Page	250	10	10K
5962-38267 01M ZX	AT28C010-25FM/883	128K x 8, 1-Mbit E^2PROM $\overline{\text{Data}}$ Polling	Byte/ Page	250	10	10K
5962-38267 03M XX	AT28C010-20BM/883	128K x 8, 1-Mbit E^2PROM $\overline{\text{Data}}$ Polling	Byte/ Page	200	10	10
5962-38267 03M YX	AT28C010-20LM/883	128K x 8, 1-Mbit E^2PROM $\overline{\text{Data}}$ Polling	Byte/ Page	200	10	10K
5962-38267 03M ZX	AT28C010-20FM/883	128K x 8, 1-Mbit E^2PROM $\overline{\text{Data}}$ Polling	Byte/ Page	200	10	10K

Table 1. Atmel SMD Part Types, Listed by Atmel Part Number

AT22V10

Generic Number	Standardized Military Drawing Number				Description	
C22V10	Drawing Number	Device Type	Case Outline	Lead Finish	Circuit Description	TPD (ns)
	5962-88670	01	L, 3	X, A, C	22-Input, 10-Output and-or-Logic Array	25
	5962-88670	02	L, 3	X, A, C	22-Input, 10-Output and-or-Logic Array	30
	5962-88670	04	L, 3	X, A, C	22-Input, 10-Output and-or-Logic Array	20
	5962-88670	05	L, 3	X, A, C	22-Input, 10-Output and-or-Logic Array	15

Atmel Cage No. 1FN41	Example: Atmel Order Number	Atmel Similar Part Number
	5962-88670 **01 LX**	AT22V10-25DM/883
	5962-88670 **01 3X**	AT22V10-25LM/883
	5962-88670 **02 LX**	AT22V10-30DM/883
	5962-88670 **02 3X**	AT22V10-30LM/883
	5962-88670 **04 LX**	AT22V10-20DM/883
	5962-88670 **04 3X**	AT22V10-20LM/883
	5962-88670 **05 LX**	AT22V10-15DM/883
	5962-88670 **05 3X**	AT22V10-15LM/883

AT22V10B

Generic Number	Standardized Military Drawing Number				Description	
C22V10B	Drawing Number	Device Type	Case Outline	Lead Finish	Circuit Description	TPD (ns)
	5962-87539	06	L, 3, X	X, A, C	22-Input, 10-Output and-or-Logic Array	10

Atmel Cage No. 1FN41	Example: Atmel Order Number	Atmel Similar Part Number
	5962-87539 **06 LX**	AT22V10B-10DM/883
	5962-87539 **06 3X**	AT22V10B-10LM/883
	5962-87539 **06 XX**	AT22V10B-10KM/883

Case Outline	
L	24DW3, 24 Lead 0.300" Wide, Windowed, Ceramic Dual Inline Package (Cerdip)
3	28LW, 28 Pad, Windowed, Ceramic Leadless Chip Carrier (LCC)
X	28KW, 28 Lead, Windowed, Ceramic J-Leaded Chip Carrier (JLCC)
Lead Finish	
X	Allows Hot Tin Dip or Gold (AU)
A	Hot Tin Dip
C	Gold (AU)

Table 1. Atmel SMD Part Types, Listed by Atmel Part Number *(continued)*

AT22V10L

Generic Number	Standardized Military Drawing Number				Description	
C22V10L	Drawing Number	Device Type	Case Outline	Lead Finish	Circuit Description	TPD (ns)
	5962-88724	01	L, 3	X, A, C	22-Input, 10-Output and-or-Logic Array	25
	5962-88724	02	L, 3	X, A, C	22-Input, 10-Output and-or-Logic Array	30
	5962-88724	04	L, 3	X, A, C	22-Input, 10-Output and-or-Logic Array	20

Atmel Cage No.	Example: Atmel Order Number	Atmel Similar Part Number
1FN41	5962-88724 01 LX	AT22V10L-25DM/883
	5962-88724 01 3X	AT22V10L-25LM/883
	5962-88724 02 LX	AT22V10L-30DM/883
	5962-88724 02 3X	AT22V10L-30LM/883
	5962-88724 04 LX	AT22V10L-20DM/883
	5962-88724 04 3X	AT22V10L-20LM/883

Case Outline	
L	24DW3, 24 Lead 0.300" Wide, Windowed, Ceramic Dual Inline Package (Cerdip)
3	28LW, 28 Pad, Windowed, Ceramic Leadless Chip Carrier (LCC)

Lead Finish	
X	Allows Hot Tin Dip or Gold (AU)
A	Hot Tin Dip
C	Gold (AU)

Table 1. Atmel SMD Part Types, Listed by Atmel Part Number *(continued)*

AT22V10 OTP

Generic Number	Standardized Military Drawing Number				Description	
C22V10 OTP	Drawing Number	Device Type	Case Outline	Lead Finish	Circuit Description	TPD (ns)
	5962-87570	01	L, 3	X, A, C	22-Input, 10-Output and-or-Logic Array	25
	5962-87570	02	L, 3	X, A, C	22-Input, 10-Output and-or-Logic Array	30
	5962-87570	04	L, 3	X, A, C	22-Input, 10-Output and-or-Logic Array	20
	5962-87570	05	L, 3	X, A, C	22-Input, 10-Output and-or-Logic Array	15

Atmel Cage No. 1FN41	Example: Atmel Order Number	Atmel Similar Part Number
	5962-87570 01 LX	AT22V10-25GM/883
	5962-87570 01 3X	AT22V10-25NM/883
	5962-87570 02 LX	AT22V10-30GM/883
	5962-87570 02 3X	AT22V10-30NM/883
	5962-87570 04 LX	AT22V10-20GM/883
	5962-87570 04 3X	AT22V10-20NM/883
	5962-87570 05 LX	AT22V10-15GM/883
	5962-87570 05 3X	AT22V10-150NM/883

AT22V10L OTP

Generic Number	Standardizedized Military Drawing Number				Description	
C22V10L OTP	Drawing Number	Device Type	Case Outline	Lead Finish	Circuit Description	TPD (ns)
	5962-89755	01	L, 3	X, A, C	22-Input, 10-Output and-or-Logic Array	25
	5962-89755	02	L, 3	X, A, C	22-Input, 10-Output and-or-Logic Array	30
	5962-89755	04	L, 3	X, A, C	22-Input, 10-Output and-or-Logic Array	20

Atmel Cage No. 1FN41	Example: Atmel Order Number	Atmel Similar Part Number
	5962-89755 01 LX	AT22V10L-25GM/883
	5962-89755 01 3X	AT22V10L-25NM/883
	5962-89755 02 LX	AT22V10L-30GM/883
	5962-89755 02 3X	AT22V10L-30NM/883
	5962-89755 04 LX	AT22V10L-20GM/883
	5962-89755 04 3X	AT22V10L-20NM/883

Case Outline	
L	24D3, 24 Lead 0.300" Wide, Non-Windowed, Ceramic Dual Inline Package (Cerdip)
3	28L, 28 Pad, Non-Windowed, Ceramic Leadless Chip Carrier (LCC)
Lead Finish	
X	Allows Hot Tin Dip or Gold (AU)
A	Hot Tin Dip
C	Gold (AU)

Table 1. Atmel SMD Part Types, Listed by Atmel Part Number *(continued)*

ATV750

Generic Number	Standardized Military Drawing Number				Description	
V750	Drawing Number	Device Type	Case Outline	Lead Finish	Circuit Description	TPD (ns)
	5962-88726	02	L, 3	X, A, C	22-Input, 10-Output and-or-Logic Array	35
	5962-88726	03	L, 3	X, A, C	22-Input, 10-Output and-or-Logic Array	25
	5962-88726	04	L, 3	X, A, C	22-Input, 10-Output and-or-Logic Array	20

Atmel Cage No.	Example: Atmel Order Number	Atmel Similar Part Number
1FN41	5962-88726 02 LX	ATV750-35DM/883
	5962-88726 02 3X	ATV750-35LM/883
	5962-88726 03 LX	ATV750-25DM/883
	5962-88726 03 3X	ATV750-25LM/883
	5962-88726 04 LX	ATV750-20DM/883
	5962-88726 04 3X	ATV750-20LM/883

ATV750L

Generic Number	Standardized Military Drawing Number				Description	
V750L	Drawing Number	Device Type	Case Outline	Lead Finish	Circuit Description	TPD (ns)
	5962-88726	06	L, 3	X, A, C	22-Input, 10-Output and-or-Logic Array	30
	5962-88726	07	L, 3	X, A, C	22-Input, 10-Output and-or-Logic Array	25

Atmel Cage No.	Example: Atmel Order Number	Atmel Similar Part Number
1FN41	5962-88726 06 LX	ATV750L-30DM/883
	5962-88726 06 3X	ATV750L-30LM/883
	5962-88726 07 LX	ATV750L-25DM/883
	5962-88726 07 3X	ATV750L-25LM/883

Case Outline	
L	24DW3, 24 Lead, 0.300" Wide, Windowed, Ceramic Dual Inline Package (Cerdip)
3	28LW, 28 Pad, Windowed, Ceramic Leadless Chip Carrier (LCC)
Lead Finish	
X	Allows Hot Tin Dip or Gold (AU)
A	Hot Tin Dip
C	Gold (AU)

Table 1. Atmel SMD Part Types, Listed by Atmel Part Number *(continued)*

ATV2500H

Generic Number	Standardized Military Drawing Number				Description	
V2500	Drawing Number	Device Type	Case Outline	Lead Finish	Circuit Description	TPD (ns)
	5962-91545	01	Q,X,Y	X, A, C	38-Input, 24-Output and-or-Logic Array	35
	5962-91545	02	Q,X,Y	X, A, C	38-Input, 24-Output and-or-Logic Array	25

Atmel Cage No.	Example: Atmel Order Number	Atmel Similar Part Number
1FN41	5962-91545 **01M QX**	ATV2500H-35DM/883
	5962-91545 **01M XX**	ATV2500H-35LM/883
	5962-91545 **01M YX**	ATV2500H-35KM/883
	5962-91545 **02M QX**	ATV2500H-25DM/883
	5962-91545 **02M XX**	ATV2500H-25LM/883
	5962-91545 **02M YX**	ATV2500H-25KM/883

ATV2500L

Generic Number	Standardized Military Drawing Number				Description	
V2500L	Drawing Number	Device Type	Case Outline	Lead Finish	Circuit Description	TPD (ns)
	5962-91545	03	Q,X,Y	X, A, C	38-Input, 24-Output and-or-Logic Array	30

Atmel Cage No.	Example: Atmel Order Number	Atmel Similar Part Number
1FN41	5962-91545 **03M QX**	ATV2500L-30DM/883
	5962-91545 **03M XX**	ATV2500L-30LM/883
	5962-91545 **03M YX**	ATV2500L-30KM/883

Case Outline	
Q	40DW6, 40 Lead, 0.600" Wide, Windowed, Ceramic Dual Inline Package (Cerdip)
X	44LW, 44 Pad, Windowed, Ceramic Leadless Chip Carrier (LCC)
Y	44KW, 44 Lead, Windowed, Ceramic J-Leaded Chip Carrier (JLCC)
Lead Finish	
X	Allows Hot Tin Dip or Gold (AU)
A	Hot Tin Dip
C	Gold (AU)

5962-87539

Atmel Order Number	Atmel Similar Part Number	Circuit Description	TPD (ns)
5962-87539 01 LX	AT22V10-25DM/883	22-Input, 10-Output and-or-Logic Array	25
5962-87539 01 3X	AT22V10-25LM/883	22-Input, 10-Output and-or-Logic Array	25
5962-87539 02 LX	AT22V10-30DM/883	22-Input, 10-Output and-or-Logic Array	30
5962-87539 02 3X	AT22V10-30LM/883	22-Input, 10-Output and-or-Logic Array	30
5962-87539 04 LX	AT22V10-20DM/883	22-Input, 10-Output and-or-Logic Array	20
5962-87539 04 3X	AT22V10-20LM/883	22-Input, 10-Output and-or-Logic Array	20
5962-87539 05 LX	AT22V10-15DM/883	22-Input, 10-Output and-or-Logic Array	15
5962-87539 05 3X	AT22V10-15LM/883	22-Input, 10-Output and-or-Logic Array	15
5962-87539 06 LX	AT22V10B-10DM/883	22-Input, 10-Output and-or-Logic Array	10
5962-87539 06 3X	AT22V10B-10LM/883	22-Input, 10-Output and-or-Logic Array	10
5962-87539 06 XX	AT22V10B-10KM/883	22-Input, 10-Output and-or-Logic Array	10

5962-88670

Atmel Order Number	Atmel Similar Part Number	Circuit Description	TPD (ns)
5962-88670 01 LX	AT22V10-25GM/883	22-Input, 10-Output and-or-Logic Array	25
5962-88670 01 3X	AT22V10-25NM/883	22-Input, 10-Output and-or-Logic Array	25
5962-88670 02 LX	AT22V10-30GM/883	22-Input, 10-Output and-or-Logic Array	30
5962-88670 02 3X	AT22V10-30NM/883	22-Input, 10-Output and-or-Logic Array	30
5962-88670 04 LX	AT22V10-20GM/883	22-Input, 10-Output and-or-Logic Array	20
5962-88670 04 3X	AT22V10-20NM/883	22-Input, 10-Output and-or-Logic Array	20
5962-88670 05 LX	AT22V10-15GM/883	22-Input, 10-Output and-or-Logic Array	15
5962-88670 05 3X	AT22V10-15NM/883	22-Input, 10-Output and-or-Logic Array	15

5962-88724

Atmel Order Number	Atmel Similar Part Number	Circuit Description	TPD (ns)
5962-88724 01 LX	AT22V10L-25DM/883	22-Input, 10-Output and-or-Logic Array	25
5962-88724 01 3X	AT22V10L-25LM/883	22-Input, 10-Output and-or-Logic Array	25
5962-88724 02 LX	AT22V10L-30DM/883	22-Input, 10-Output and-or-Logic Array	30
5962-88724 02 3X	AT22V10L-30LM/883	22-Input, 10-Output and-or-Logic Array	30
5962-88724 04 LX	AT22V10L-20DM/883	22-Input, 10-Output and-or-Logic Array	20
5962-88724 04 3X	AT22V10L-20LM/883	22-Input, 10-Output and-or-Logic Array	20

5962-88726

Atmel Order Number	Atmel Similar Part Number	Circuit Description	TPD (ns)
5962-88726 02 LX	ATV750-35DM/883	22-Input, 10-Output and-or-Logic Array	35
5962-88726 02 3X	ATV750-35LM/883	22-Input, 10-Output and-or-Logic Array	35
5962-88726 03 LX	ATV750-25DM/883	22-Input, 10-Output and-or-Logic Array	25
5962-88726 03 3X	ATV750-25LM/883	22-Input, 10-Output and-or-Logic Array	25
5962-88726 04 LX	ATV750-20DM/883	22-Input, 10-Output and-or-Logic Array	20
5962-88726 04 3X	ATV750-20LM/883	22-Input, 10-Output and-or-Logic Array	20
5962-88726 06 LX	ATV750L-30DM/883	22-Input, 10-Output and-or-Logic Array	30
5962-88726 06 3X	ATV750L-30LM/883	22-Input, 10-Output and-or-Logic Array	30
5962-88726 07 LX	ATV750L-25DM/883	22-Input, 10-Output and-or-Logic Array	25
5962-88726 07 3X	ATV750L-25LM/883	22-Input, 10-Output and-or-Logic Array	25

Table 2. Atmel SMD Part Types, Listed by SMD Number *(continued)*

5962-89755			
Atmel Order Number	Atmel Similar Part Number	Circuit Description	TPD (ns)
5962-89755 01 LX	AT22V10L-25GM/883	22-Input, 10-Output and-or-Logic Array	25
5962-89755 01 3X	AT22V10L-25NM/883	22-Input, 10-Output and-or-Logic Array	25
5962-89755 02 LX	AT22V10L-30GM/883	22-Input, 10-Output and-or-Logic Array	30
5962-89755 02 3X	AT22V10L-30NM/883	22-Input, 10-Output and-or-Logic Array	30
5962-89755 04 LX	AT22V10L-20GM/883	22-Input, 10-Output and-or-Logic Array	20
5962-89755 04 3X	AT22V10L-20NM/883	22-Input, 10-Output and-or-Logic Array	20

5962-91545			
Atmel Order Number	Atmel Similar Part Number	Circuit Description	TPD (ns)
5962-91545 01M QX	ATV2500H-35DM/883	38-Input, 24-Output and-or-Logic Array	35
5962-91545 01M XX	ATV2500H-35LM/883	38-Input, 24-Output and-or-Logic Array	35
5962-91545 01M YX	ATV2500H-35KM/883	38-Input, 24-Output and-or-Logic Array	35
5962-91545 02M QX	ATV2500H-25DM/883	38-Input, 24-Output and-or-Logic Array	25
5962-91545 02M XX	ATV2500H-25LM/883	38-Input, 24-Output and-or-Logic Array	25
5962-91545 02M YX	ATV2500H-25KM/883	38-Input, 24-Output and-or-Logic Array	25
5962-91545 03M QX	ATV2500L-30DM/883	38-Input, 24-Output and-or-Logic Array	30
5962-91545 03M XX	ATV2500L-30LM/883	38-Input, 24-Output and-or-Logic Array	30
5962-91545 03M YX	ATV2500L-30KM/883	38-Input, 24-Output and-or-Logic Array	30

Section 8 Standard Package Outlines

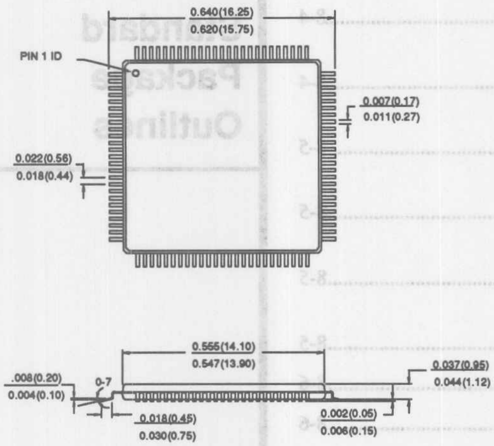
Standard Package Outlines

Each Atmel data sheet includes an Ordering Information Section which specifies the package types available. This section provides size specifications and outlines for all package types.[1]

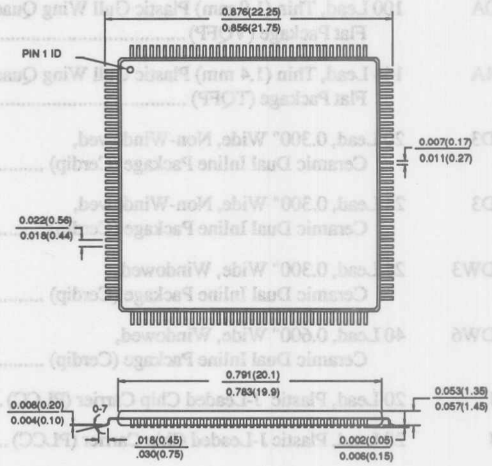
Note: 1. Dimensions shown do not include lead plating or mold flash.

8

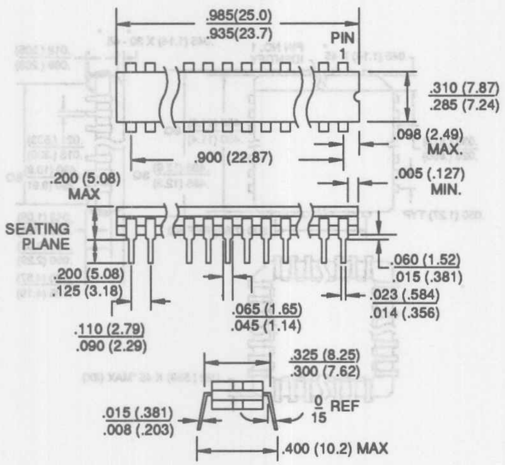
100A, 100 Lead, Very Thin (1.0 mm) Plastic Gull Wing Quad Flat Package (VQFP)
Dimensions in Inches and (Millimeters)



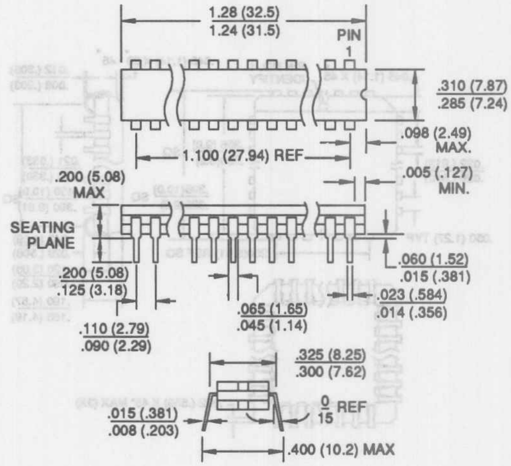
144A, 144 Lead, Thin (1.0 mm) Plastic Gull Wing Quad Flat Package (TQFP)
Dimensions in Inches and (Millimeters)



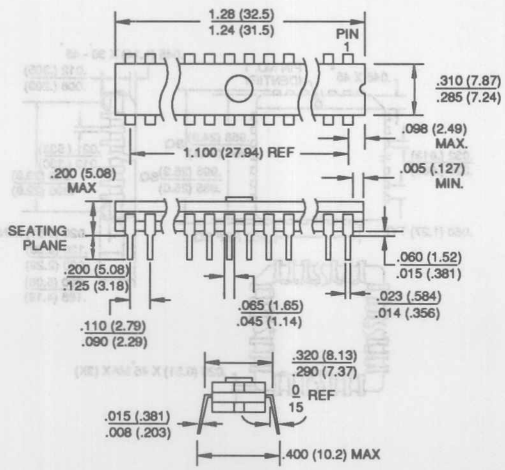
20D3, 20 Lead, 0.300" Wide, Non-Windowed, Ceramic Dual Inline Package (Cerdip)
Dimensions in Inches and (Millimeters)
MIL-STD-1835 D-8 CONFIG A



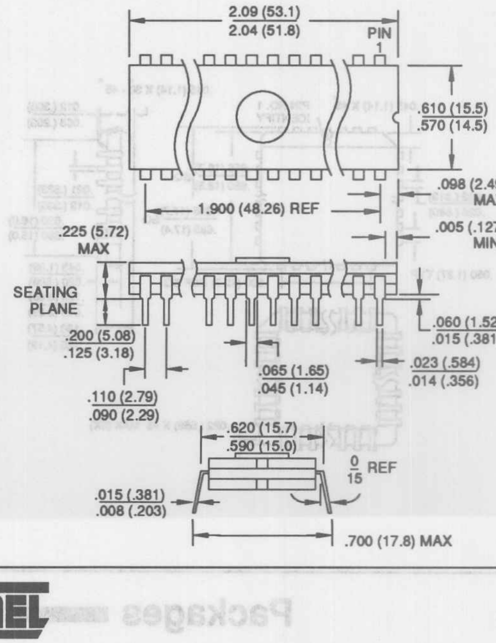
24D3, 24 Lead, 0.300" Wide, Non-Windowed Ceramic Dual Inline Package (Cerdip)
Dimensions in Inches and (Millimeters)
MIL-STD-1835 D-9 CONFIG A



24DW3, 24 Lead, 0.300" Wide, Windowed, Ceramic Dual Inline Package (Cerdip)
Dimensions in Inches and (Millimeters)
MIL-STD-1835 D-9 CONFIG A

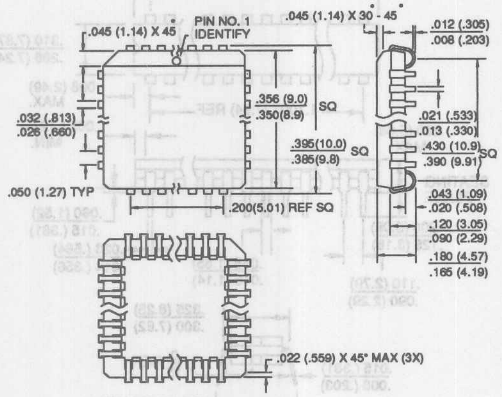


40DW6, 40 Lead, 0.600" Wide, Windowed, Ceramic Dual Inline Package (Cerdip)
Dimensions in Inches and (Millimeters)
MIL-STD-1835 D-5 CONFIG A

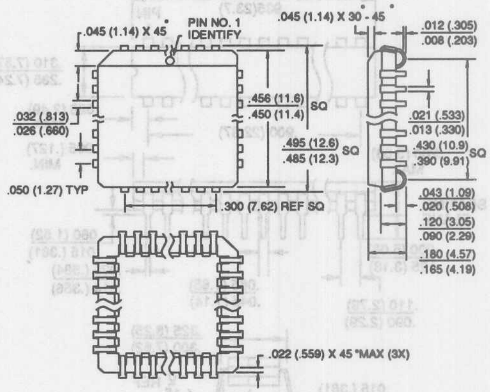


8

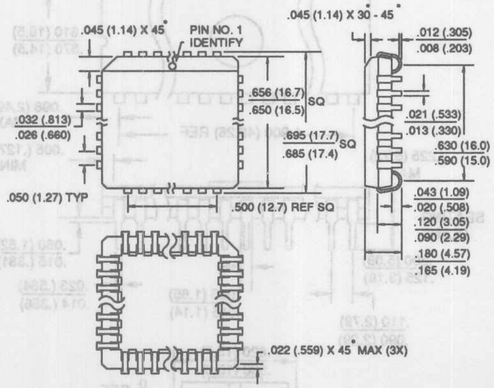
20J, 20 Lead, Plastic J-Leaded Chip Carrier (PLCC)
Dimensions in Inches and (Millimeters)



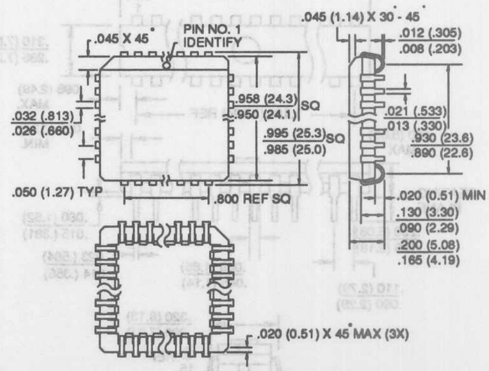
28J, 28 Lead, Plastic J-Leaded Chip Carrier (PLCC)
Dimensions in Inches and (Millimeters)
JEDEC OUTLINE M0-047 AB



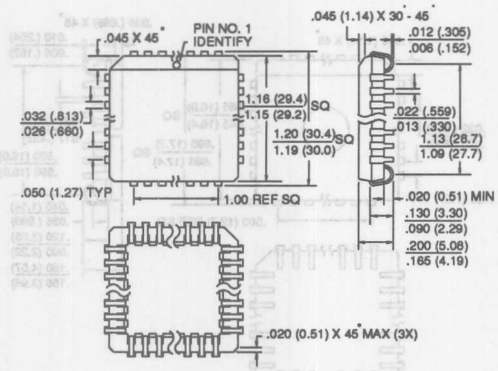
44J, 44 Lead, Plastic J-Leaded Chip Carrier (PLCC)
Dimensions in Inches and (Millimeters)
JEDEC OUTLINE MO-047 AC



68J, 68 Lead, Plastic J-Leaded Chip Carrier (PLCC)
Dimensions in Inches and (Millimeters)
JEDEC OUTLINE MO-047 AE

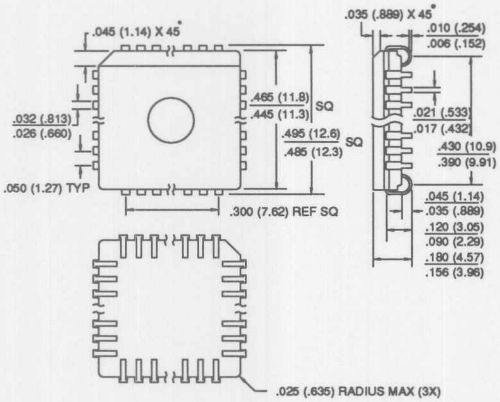


84J, 84 Lead, Plastic J-Leaded Chip Carrier (PLCC)
Dimensions in Inches and (Millimeters)
JEDEC OUTLINE MO-47 AF

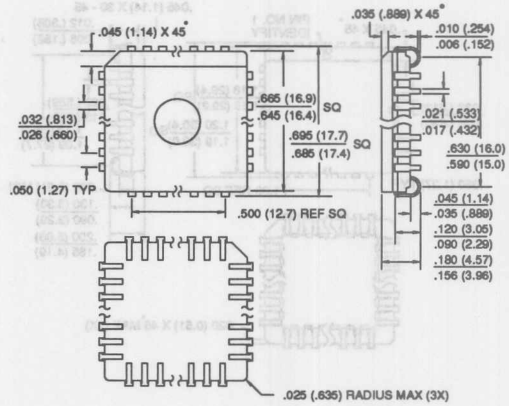


8

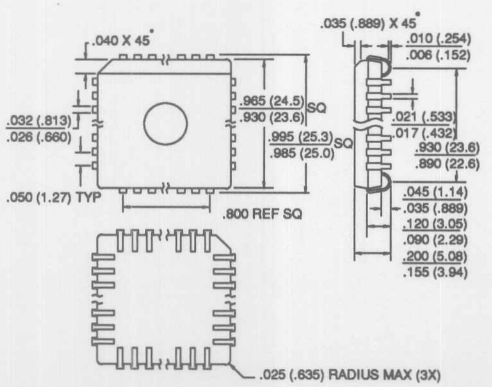
28KW, 28 Lead, Windowed,
Ceramic J-Leaded Chip Carrier (JLCC)
Dimensions in Inches and (Millimeters)
MIL-STD-1835 C-J7
JEDEC OUTLINE MO-087 AA



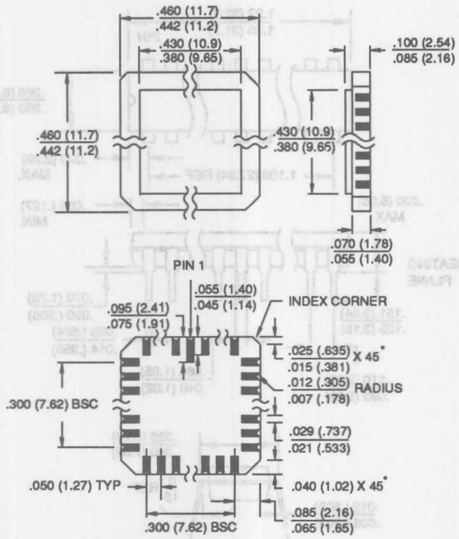
44KW, 44 Lead, Windowed,
Ceramic J-Leaded Chip Carrier (JLCC)
Dimensions in Inches and (Millimeters)
MIL-STD-1835 C-J1



68KW, 68 Lead, Windowed,
Ceramic J-Leaded Chip Carrier (JLCC)
Dimensions in Inches and (Millimeters)
MIL-STD-1835 C-J2
JEDEC OUTLINE MO-087 AD

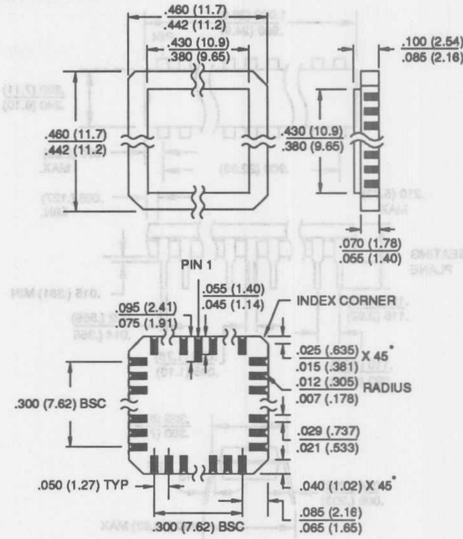


20L, 20 Pad, Non-Windowed,
Ceramic Leadless Chip Carrier (LCC)
Dimensions in Inches and (Millimeters)*
MIL-STD-1835 C-4



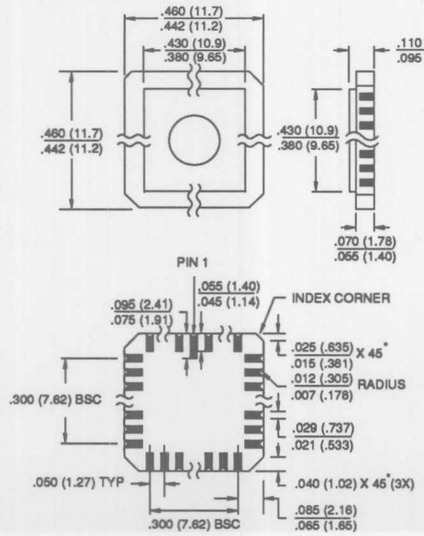
*Ceramic lid standard unless specified.

28L, 28 Pad, Non-Windowed,
Ceramic Leadless Chip Carrier (LCC)
Dimensions in Inches and (Millimeters)*
MIL-STD-1835 C-4



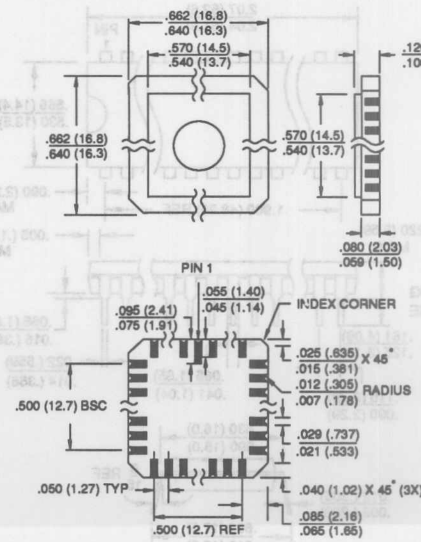
*Ceramic lid standard unless specified.

28LW, 28 Pad, Windowed,
Ceramic Leadless Chip Carrier (LCC)
Dimensions in Inches and (Millimeters)*
MIL-STD-1835 C-4



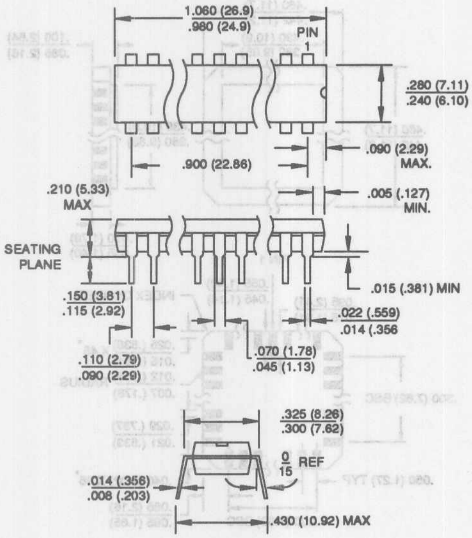
*Ceramic lid standard unless specified.

44LW, 44 Pad, Windowed,
Ceramic Leadless Chip Carrier (LCC)
Dimensions in Inches and (Millimeters)*
MIL-STD-1835 C-5

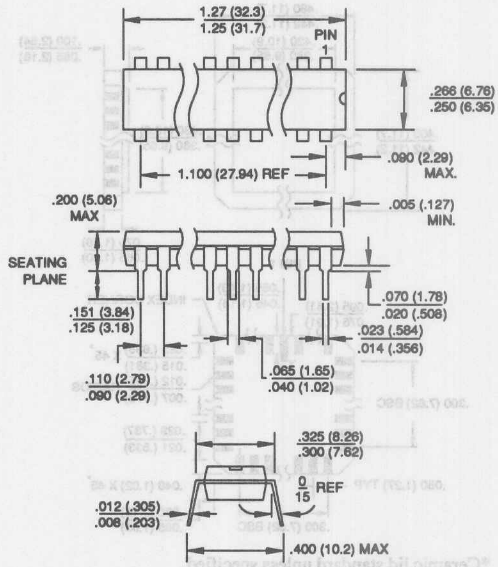


*Ceramic lid standard unless specified.

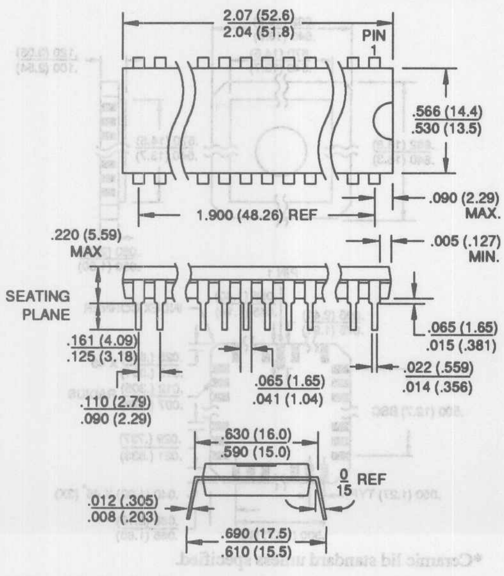
20P3, 20 Lead, 0.300" Wide, Plastic Dual Inline Package (PDIP) Dimensions in Inches and (Millimeters)



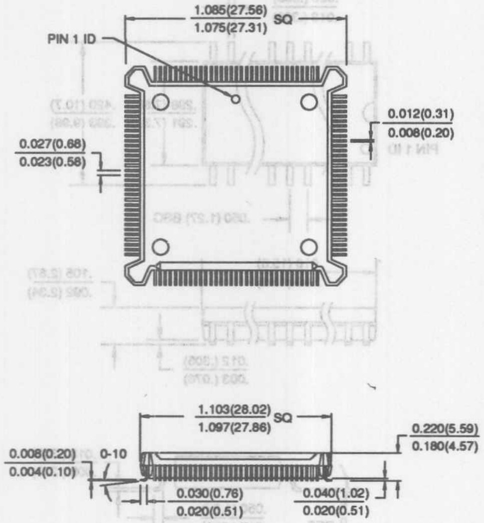
24P3, 24 Lead, 0.300" Wide, Plastic Dual Inline Package (PDIP) Dimensions in Inches and (Millimeters)



40P6, 40 Lead, 0.600" Wide, Plastic Dual Inline Package (PDIP) Dimensions in Inches and (Millimeters)

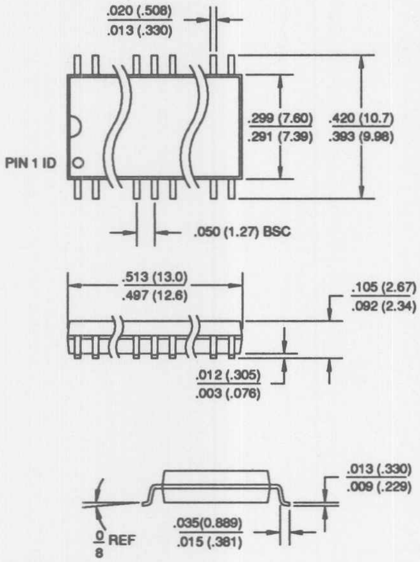


132Q, 132 Lead, Plastic Gull Wing Quad Flat Package (PQFP)
Dimensions in Inches and (Millimeters)

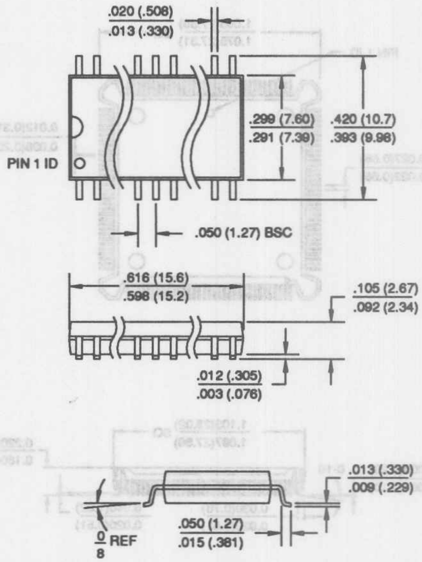


8

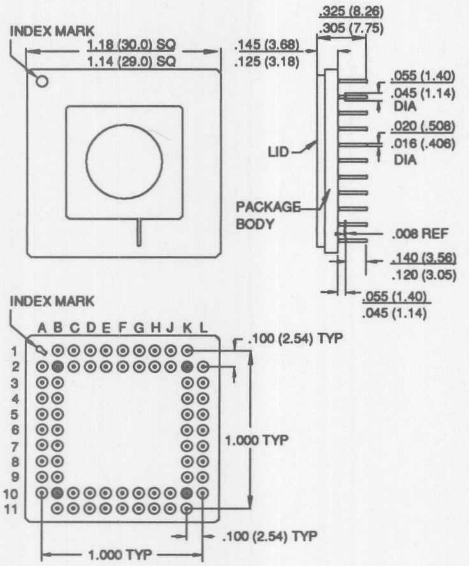
20S, 20 Lead, 0.300" Wide,
Plastic Gull Wing Small Outline (SOIC)
Dimensions in Inches and (Millimeters)



24S, 24 Lead, 0.300" Wide,
Plastic Gull Wing Small Outline (SOIC)
Dimensions in Inches and (Millimeters)



68UW, 68 Pin, Windowed, Ceramic Pin Grid Array (PGA)
Dimensions in Inches and (Millimeters)



8

Thermal Specifications

Thermal Characteristics of Atmel Packages

The thermal performance of the semiconductor package is a very important consideration for the board designer. The reliability and functional life of the device is directly related to its junction operating temperature. As the temperature of the device increases, the stability of its junctions decline, as does its reliable life. The thermal performance is also important to the board design, because it may limit the board density, or dictate the board location of high power-dissipating devices, or require expensive cooling methods for the system. As devices have become more complex and boards have become denser, the need to account for the thermal characteristics of packages has shifted from being a minor consideration to being a necessary consideration.

The thermal performance of a package is measured by its ability to dissipate the power required by the device into its surroundings. The electrical power drawn by the device generates heat on the top surface of the die. This heat is conducted through the package to the surface and then transferred to the surrounding air by convection. Each heat transfer step has a corresponding *resistance* to the heat flow, which is given the value θ, the thermal resistance coefficient. Subscripts are added to the coefficient to specify the two points that the heat is transferred between. Commonly used coefficients are θ_{JA} (junction to ambient air), θ_{JC} (junction to package case), and θ_{CA} (case to ambient air).

An electrical analogy can be made, as shown in the figure below, to illustrate the heat flow of a package. The heat transfer can be characterized mathematically by the following equation,

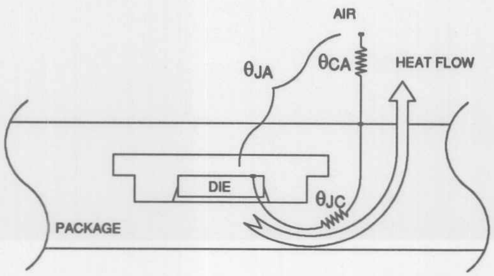
$$Tj - Ta = P \times \theta_{JA}$$

where,

P = *Device operating power [watts]*
Tj = *Temperature of a junction on the device [℃]*
Ta = *Temperature of the surrounding ambient air [℃]*

Two conclusions can be made after examining this analogy. First, the lower the value of θ_{JA}, the better the heat dissipation of the package. Secondly, the value of θ_{JA} is directly dependent upon both the conductive (θ_{JC}) and convective (θ_{CA}) properties of the package. θ_{JC} is a function of the package material, the adhesion between the package materials, and device size. θ_{CA} is a function of the package size and configuration, package mounting method, and air flow across the package. Lower θ_{JA} values can be achieved by specifying ceramic packages instead of plastic packages, choosing larger packages, or improving air flow across the package.

The thermal resistance values of Atmel standard packages are listed on the following page. The figures shown are maximum values for θ, typical values are lower dependent upon the device type.



Thermal Resistance Coefficients

		θ_{JC} [°C/W]	θ_{JA} [°C/W] Airflow = 0 ft/min
Ceramic DIP	24D3/DW3	9	65
	40DW6	7	40
Plastic DIP	24P3	22	82
	40P6	30	68
Leadless Chip Carrier (LCC)	28L/LW	12	68
	44LW	8-10	60
Plastic Leaded Chip Carrier (PLCC)	28J	16	60
	44J	14	50
	68J	10	45
J-Leaded Chip Carrier (JLCC)	28K/KW	16	72
	32K/KW	16	72
	44K/KW	16	68
	68KW	10-14	47
Ceramic Pin Grid Array (PGA)	68UW	4-6	30-40

Flash PEROMs

Part Number	Organization	Speeds	Description
Low Voltage			
AT29LV256	32K x 8	200-250 ns	256K, 3-Volt Programmable, Erasable ROM
AT29LV512	64K x 8	200-250 ns	512K, 3-Volt Programmable, Erasable ROM
AT29LV010	128K x 8	200-250 ns	1-Mbit, 3-Volt Programmable, Erasable ROM
AT29LV020	256K x 8	200-250 ns	2-Mbit, 3-Volt Programmable, Erasable ROM
AT29LV040	512K x 8	200-250 ns	4-Mbit, 3-Volt Programmable, Erasable ROM
Standard			
AT29C256	32K x 8	90-250 ns	256K, 5-Volt Programmable, Erasable ROM
AT29C257	32K x 8	90-250 ns	256K, 5-Volt Programmable, Erasable ROM
AT29C512	64K x 8	90-200 ns	512K, 5-Volt Programmable, Erasable ROM
AT29C1024	64K x 16	90-200 ns	1-Mbit, 5-Volt Programmable, Erasable ROM
AT29C010	128K x 8	90-200 ns	1-Mbit, 5-Volt Programmable, Erasable ROM
AT29C020	256K x 8	100-200 ns	2-Mbit, 5-Volt Programmable, Erasable ROM
AT29C040	512K x 8	120-250 ns	4-Mbit, 5-Volt Programmable, Erasable ROM

Serial EEPROMs

Part Number	Organization	V_{CC}	Description
AT24C01	128 x 8	1.8, 2.5, 2.7, 5.0 V	2-Wire, 1K Serial E^2PROM
AT24C01A	128 x 8	1.8, 2.5, 2.7, 5.0 V	2-Wire, 1K Serial E^2PROM
AT24C02	256 x 8	1.8, 2.5, 2.7, 5.0 V	2-Wire, 2K Serial E^2PROM
AT24C04	512 x 8	1.8, 2.5, 2.7, 5.0 V	2-Wire, 4K Serial E^2PROM
AT24C08	1024 x 8	1.8, 2.5, 2.7, 5.0 V	2-Wire, 8K Serial E^2PROM
AT24C16	2048 x 8	1.8, 2.5, 2.7, 5.0 V	2-Wire, 16K Serial E^2PROM
AT24C164	2048 x 8	1.8, 2.5, 2.7, 5.0 V	2-Wire, 16K Serial E^2PROM
AT93C46	64 x 16 / 128 x 8	1.8, 2.5, 2.7, 5.0 V	3-Wire, 1K Serial E^2PROM
AT93C56	128 x 16 / 256 x 8	2.7, 5.0 V	3-Wire, 2K Serial E^2PROM
AT93C57	128 x 16 / 256 x 8	2.7, 5.0 V	3-Wire, 2K Serial E^2PROM
AT93C66	256 x 16 / 512 x 8	2.7, 5.0 V	3-Wire, 4K Serial E^2PROM
AT59C11	64 x 16 / 128 x 8	2.7, 5.0 V	4-Wire, 1K Serial E^2PROM
AT59C22	128 x 16 / 256 x 8	2.7, 5.0 V	4-Wire, 2K Serial E^2PROM
AT59C13	256 x 16 / 512 x 8	2.7, 5.0 V	4-Wire, 4K Serial E^2PROM

Parallel EEPROMs

Part Number	Organization	Speeds	Description
High Speed			
AT28HC64B	8K x 8	55-120 ns	64K E^2PROM with 64-Byte Page & Software Data Protection
AT28HC256,L	32K x 8	70-120 ns	256K E^2PROM with 64-Byte Page & Software Data Protection
AT28HC256E,LE	32K x 8	70-120 ns	256K E^2PROM with Extended Endurance, Standard & Low Power
AT28HC256F,LF	32K x 8	70-120 ns	256K E^2PROM with Fast Write, Standard & Low Power
Low Voltage			
AT28LV64	8K x 8	200-300 ns	64K E^2PROM
AT89LV64B	8K x 8	200-300 ns	64K E^2PROM with 64-Byte Page & Software Data Protection
AT28LV256	32K x 8	200-300 ns	256K E^2PROM with 64-Byte Page & Software Data Protection
Standard			
AT28C04	512 x 8	150-250 ns	4K E^2PROM
AT28C04E	512 x 8	150-250 ns	4K E^2PROM with Extended Endurance
AT28C04F	512 x 8	150-250 ns	4K E^2PROM with Fast Write
AT28C16	2K x 8	150-250 ns	16K E^2PROM
AT28C16E	2K x 8	150-250 ns	16K E^2PROM with Extended Endurance
AT28C16F	2K x 8	150-250 ns	16K E^2PROM with Fast Write
AT28C17	2K x 8	150-250 ns	16K E^2PROM with Ready/$\overline{\text{Busy}}$
AT28C17E	2K x 8	150-250 ns	16K E^2PROM with Ready/$\overline{\text{Busy}}$ & Extended Endurance
AT28C17F	2K x 8	150-250 ns	16K E^2PROM with Ready/Busy & Fast Write
AT28C64	8K x 8	120-350 ns	64K E^2PROM
AT28C64E	8K x 8	120-350 ns	64K E^2PROM with Extended Endurance
AT28C64F	8K x 8	120-350 ns	64K E^2PROM with Fast Write
AT28C64X	8K x 8	150-450 ns	64K E^2PROM without Ready-$\overline{\text{Busy}}$
AT28C64B	8K x 8	150-250 ns	64K E^2PROM with 64-Byte Page, Software Data Protection & Extended Endurance
AT28C256	32K x 8	150-350 ns	256K E^2PROM with 64-Byte Page & Software Data Protection
AT28C256E	32K x 8	150-350 ns	256K E^2PROM with Extended Endurance
AT28C256F	32K x 8	150-350 ns	256K E^2PROM with Fast Write
AT28C1024	64K x 16	120-250 ns	1-Mbit E^2PROM
AT28C010	128K x 8	120-250 ns	1-Mbit E^2PROM with 128-Byte Page
AT28C010E	128K x 8	120-250 ns	1-Mbit E^2PROM with 128-Byte Page & Extended Endurance

EPROMs

Part Number	Organization	Speeds	Description
Low Voltage			
AT27LV256R	32K x 8	150-250 ns	256K 3-Volt EPROM
AT27LV512R	64K x 8	150-250 ns	512K 3-Volt EPROM
AT27LV1024	64K x 16	150-250 ns	1-Mbit, 3-Volt EPROM
AT27LV010	128K x 8	150-250 ns	1-Mbit, 3-Volt EPROM
AT27LV020	256K x 8	200-300 ns	2-Mbit, 3-Volt EPROM
AT27LV4096	256K x 16	250-300 ns	4-Mbit, 3-Volt EPROM
AT27LV040	512K x 8	200-300 ns	4-Mbit, 3-Volt EPROM
AT27LV080	1024K x 8	250-300 ns	8-Mbit, 3-Volt EPROM
Standard			
AT27C256R	32K x 8	55-200 ns	256K EPROM
AT27C512R	64K x 8	70-200 ns	512K EPROM
AT27C1024	64K x 16	70-200 ns	1-Mbit EPROM
AT27C010,L	128K x 8	70-200 ns	1-Mbit EPROM, Standard & Low Power
AT27C020	256K x 8	85-200 ns	2-Mbit EPROM
AT27C4096	256K x 16	100-200 ns	4-Mbit EPROM
AT27C040	512K x 8	100-200 ns	4-Mbit EPROM
AT27C080	1024K x 8	100-200 ns	8-Mbit EPROM

PROMs

Part Number	Organization	Speeds	Description
AT27HC641R	8K x 8	35-90 ns	High Speed 64K Reprogrammable [UV]PROM
AT27HC642R	8K x 8	35-90 ns	High Speed 64K Reprogrammable [UV]PROM

Flash Memory Card

Part Number	Organization	V_{CC}	Description
AT5FC001	1 Mbyte	5.0 V	501 PCMCIA/Compatible Flash Memory Card
AT5FC002	2 Mbyte	5.0 V	501 PCMCIA/Compatible Flash Memory Card
AT3FC001	1 Mbyte	3.3 V	501 PCMCIA/Compatible Flash Memory Card
AT3FC002	2 Mbyte	3.3 V	501 PCMCIA/Compatible Flash Memory Card

Logic

Part Number	Speeds	Description
AT40281	16-40 MHz	80386SX PC/AT Core Logic Controller, with Posted-Write Cache
AT40283	16-33 MHz	80386SX PC/AT Core Logic Controller
AT40285	16-40 MHz	80386SX/486SLC/486SLC2 PC/AT Core Logic Controller
AT40391B	25-40 MHz	80386DX PC/AT System & Cache Controller
AT40392	25-50 MHz	80386DX PC/AT Memory Controller
AT40493	25-50 MHz	80486 PC/AT System & Cache Controller
AT40495	25-50 MHZ	80486 PC/AT System & Cache Controller
AT40498	25-50 MHz	80486 Core Logic Controller
AT40957	33-66 MHz	EISA/ISA PC/AT Integrated System Peripheral
AT40958	33-66 MHz	EISA/ISA PC/AT Bus Controller & Data Buffer
AT40959	33-66 MHz	EISA/ISA PC/AT DRAM & Cache Controller
AT43216	40 MHz	Single-Chip Fast SCSI Controller

Smart Card ICs

Part Number	Memory Size	Description
AT24C01	1K	2-Wire, Serial E^{2}PROM
AT24C02	2K	2-Wire, Serial E^{2}PROM
AT24C04	4K	2-Wire, Serial E^{2}PROM
AT24C08	8K	2-Wire, Serial E^{2}PROM
AT24C16	16K	2-Wire, Serial E^{2}PROM
AT93C46	1K	3-Wire, Serial E^{2}PROM
AT93C56	2K	3-Wire, Serial E^{2}PROM
AT93C57	2K	3-Wire, Serial E^{2}PROM
AT93C66	4K	3-Wire, Serial E^{2}PROM
AT88SC06	104 x 1	104-Bit Serial E^{2}PROM with Security
AT88SC101	1024 x 1	1K Serial E^{2}PROM with Security, 1 Memory Zone, 1024 Bits
AT88SC102	1024 x 1	1K Serial E^{2}PROM with Security, 2 Memory Zones, 512 Bits Each
AT88SC200	2048 x 1	2K Serial E^{2}PROM with Gate Array
AT88SC54C	16K x 8	Secure 80C31 Controller with 8K Flash, 8K E^{2}PROM, Public Key Math Coprocessor
RF ID ASICs	Up to 16K x 1	Analog, Digital & Memory on Single-Chip ASIC

Microcontroller

Part Number	Memory Size	Description
AT89C51	4K x 8	80C31 Microcontroller with 4K Bytes Flash PEROM
AT89LV51	4K x 8	2.7-Volt, 80C31 Microcontroller with 4K Bytes Flash PEROM

Mixed Signal

Part Number	Frequency	Description
AT76C10	4 kHz	Programmable Phone Line Equalizer
AT76C10E	4 kHz	Programmable Phone Line Equalizer with On-Board E^{2}PROM
AT76C176	40-66 MHz	Triple 6-Bit Color Palette DAC
AT76C176A	50-135 MHz	Triple 6-Bit Color Palette DAC with Power-Down
AT76C120	96 kHz	Dual Channel 16/18-Bit A/D Converter